UNIVERSITY OF HELSINKI
Faculty of Biological and Environmental Sciences
Department of Genetics

Henrikki Almusa

Genome analysis pipeline for next-generation sequencing

Henrikki Almusa
February 14, 2011

Supervisor: Janna Saarela, PhD, MD

HELSINGIN YLIOPISTO
HELSINGFORS UNIVERSITET
UNIVERSITY OF HELSINKI

Tiedekunta/Osasto Fakultet/Sektion – Faculty	Laitos Institution – Department	
Biotieteellinen tiedekunta	Genetiikan laitos	
Tekijä Författare – Author		
Henrikki Almusa		
Työn nimi Arbetets titel – Title		
Genomin analyysiohjelmisto toisen sukupolven sekvensaattoreille		
Oppiaine Läroämne – Subject		
Biotekniikka		
Työn laji Arbetets art – Level	Aika Datum – Month and year	Sivumäärä Sidoantal – Number of pages
Pro gradu tutkielma	14.02.2011	70

Tiivistelmä Referat – Abstract

Toisen sukupolven sekvensointilaitteet tuottavat huomattavan suuren määrän sekvenssiä lyhyessä ajassa verrattuna ensimmäisen sukupolven laitteisiin. Taustaosassa annetaan yleiskuva eri toisen sukupolven sekvensaattorien toimintamenetelmistä. Tarkemmin paneudutaan Illumina Genome Analyzer II laitteeseen, jolla tuotettiin sekvenssit tätä tutkielmaa varten.

Tällä tutkielmalla on kaksi tavoitetta. Ensimmäinen tavoite on tehdä analyysiohjelmisto genomista sekvensointia varten. Toinen tavoite on käyttää tätä ohjelmistoa vertailemaan ihmisen kaikkien geenien eksonien sekvensointimenetelmiä kahdelta eri valmistajalta, Roche Nimblegeniltä ja Agilentilta.

Materiaali ja metodi osassa kuvataan ohjelmiston toiminta tarkemmin. Kaikista ohjelmistolle annettavista tiedostoista on kuvaus sekä esimerkki. Ohjelmisto linjaa sekvensointilaitteen tuottamat lyhyet sekvenssit vertailugenomia vastaan, etsii linjauksesta varioivia kohtia ja antaa tietoa miten tuotetut sekvenssit kattavat suunnitellut genomialueet. Lisäksi tulostiedostot sisältävät sekvenssiparien poikkeavuuksia, suurempien sekvenssin lisäyksen tai poiston auheuttavia muutoksia ja yritetään yhdistellä ei linjattuja sekvenssejä isommiksi osiksi. Sekvensointi paketit eri valmistajilta myös esitellään ja tehdyt muutokset valmistajien suosittamiin ohjeisiin listataan. Viimeisenä osana käydään läpi työssä käytettyjen ohjelmistoajoille annetut tiedostot sekä muut niihin liittyvät muutokset.

Analyysiohjelmiston tuloksena tuotetaan perustason analyysi sekvenssoinnista sekä sen laadusta. Kaikki tulostiedostot selitetään käyttäjälle. Tulosten perusteella voi käyttäjä sitten tehdä syvempää analyysia oman projektinsa tarpeiden mukaan.

Eksomivertailussa Nimblegenin sekvensointimenetelmä näyttäisi olevan parempi kohdealuen sekvensointiin sekä omalla että itsenäisellä aluemäärittelyllä. Agilentin menetelmä tuotti laajemman yksinkertaisen sekvenssipeiton ihmisgenomin eksoneihin, mikä kuitenkin on liian vähäinen luotettavaa variaatioiden tunnistamista varten. Nimblegenin menetelmä sen sijaan kattoi enemmän tavoiteltuja sekvenssialueita kun vaadittiin variaatioiden tunnistamiseen riittävä sekvenssipeitto (vähintään 10 sekvenssiä). Nimblegenin menetelmä tuotti myös vähemmän virheellisiä sekvenssipoikkeavuuksia.

Avainsanat – Nyckelord – Keywords
NGS, Illumina GA II, eksomisekvensointi

Säilytyspaikka – Förvaringställe – Where deposited

Muita tietoja – Övriga uppgifter – Additional information

HELSINGIN YLIOPISTO
HELSINGFORS UNIVERSITET
UNIVERSITY OF HELSINKI

Tiedekunta/Osasto Fakultet/Sektion – Faculty	Laitos Institution – Department	
Faculty of Biological and Environmental Sciences	Department of Genetics	
Tekijä Författare – Author		
Henrikki Almusa		
Työn nimi Arbetets titel – Title		
Genome analysis pipeline for next-generation sequencing		
Oppiaine Läroämne – Subject		
Biotechnology		
Työn laji Arbetets art – Level	Aika Datum – Month and year	Sivumäärä Sidoantal – Number of pages
Master's Thesis	14.02.2011	70

Tiivistelmä Referat – Abstract

The next-generation sequencing (NGS) platforms create a large amount of sequence in short amount of time, when compared to first generation sequencers. An overview of the NGS platforms is provided with more in-depth look into Illumina Genome Analyzer II as that is used to create the data for the thesis.

There were two main aims in this thesis. First, to create a pipeline which can be used to analyse genomic sequencing. Second, to use the pipeline to compare whole human exome capture methods from two manufacturers, Roche Nimblegen and Agilent.

The pipeline is describe in detail in material and methods. All the inputs for the pipeline are described and examples shown. In the pipeline the given sequences are first aligned against the reference genome. Then various separate analysis is performed to retrieve variants and coverage of the sequencing. Supplementary results include paired-end anomalies, larger insertion and deletion polymorphisms and assembly of non-aligned sequences. The two capture methods are also described and changes to the manufacturers' recommended protocols are listed. Finally, the section has the options and various inputs used in the pipeline runs of the exome data.

The results of the pipeline is a basic level of analysis of the sequencing as well as various graphs showing the quality of the run. All the output files intended for user are described. By using the results of the pipeline, the user can do more in-depth analysis as required by the project.

When comparing the two exome capture methods, the Nimblegen capture was shown to be more efficient in capturing the CCDS exome. While the Agilent capture kit provided better one fold coverage over the exome, higher fold coverage (over 10 fold), which is required for reliable variant calling in next-generation sequencing, was better reached using the Nimblegen capture kit. Also, significantly fewer false positive paired-end anomalies were observed in the library created by using the Nimblegen capture.

Avainsanat – Nyckelord – Keywords
Next-generation sequencing, NGS, Illumina GA II, pipeline, exome

Säilytyspaikka – Förvaringställe – Where deposited

Muita tietoja – Övriga uppgifter – Additional information

Contents

Abbreviations

API	Application programming interface
CCDS	Consensus coding sequence
CNV	Copy-number variant
CPU	Central processing unit
CRT	Cyclic reversible termination
emPCR	Emulsion polymerase chain reaction
exome	All exons in an organism
FIMM	Institute of Molecular Medicine Finland
GA II	Genome Analyzer II
indel	Insertion or deletion polymorphism in sequence
LM-PCR	Ligation mediated polymerase chain reaction
NGS	Next generation sequencing
qPCR	Quantitative real-time polymerase chain reaction
PCR	Polymerase chain reaction
SBL	Sequencing by ligation
SNA	Single-nucleotide addition
SNP	Single nucleotide polymorphism
VCP	Variant calling pipeline

Chapter 1

Introduction

The thesis is about creating a variant calling pipeline for paired-end sequencing data from Illumina Genome Analyzer II using paired reads and then using it for comparison of two exome analysis kits. However the pipeline would work for any other sequencer using paired reads; so it's not attached as such to any particular sequencing platform.

1.1 Next Generation Sequencing

Sanger sequencing was invented in the 1977 by Sanger and Coulson (Sanger et al., 1977). The automated version of Sanger sequencing is considered to be first generation sequencing with the ideas springing up at the end of 1980 (Zimmermann et al., 1988). In last few years next-generation sequencing (NGS) platforms have taken over most of the large scale sequencing tasks. Sanger sequencing is still considered to be the gold standard, but the costs are one dollar per kilobase which means that most laboratories cannot afford to use it for large scale sequencing needs. The speed of data generation is also too slow for a large scale sequencing, as the latest automated Sanger sequencers create about 14 million base pairs of sequence per week if run 24 hours per day.

NGS platforms create a lot more data at lower cost per nucleotide than first generation machines. On average, the current available platforms create 25 gigabases of sequence per week. The latest machines increase the data generation ten fold. NGS platforms have enabled many of the older methodologies to be replaced by sequencing due to be widening of scope or by enhancing accuracy, invention of whole new areas and required development of new algorithms to handle the amount of data produced.

There are six manufacturers for NGS sequencers, most used ones being: Illumina with Illumina Genome Analyzer, Roche with 454 and Applied Biosystems with SOLiD.

There is no single chemistry by which the NGS machines work as there was with first generation machines. They all share some general steps but proceed to do them in a variety of ways (Voelkerding et al., 2009; Metzker, 2010). The steps are

1. Template Preparation
2. Sequencing and Detection
3. Data Analysis

1.1.1 Template Preparation

In template preparation the target DNA is first fragmented into smaller pieces, which are called fragment template or mate-pair template, and then attached to a solid surface. This immobilisation allows thousands to billions of sequencing events to be performed simultaneously as they are physically separated from each other. However in the case of Pacific Biosciences an enzyme instead of DNA fragments is attached to the solid surface.

There are two methods for template preparation.

Clonally amplified templates

The fragment template used in this method is amplified to have millions of copies. These can be affixed to beads or a solid surface, depending on the platforms used. The most common being emPCR and solid-phase amplification.

In emPCR, the fragments are captured to beads in which the amplification is done. After that the beads are attached to support which varies between the manufacturers. In solid-phase amplification, the fragments are attached to a solid-surface directly and amplified. Clonally amplified templates have problems with dephasing, where some probes get out of sync in sequencing, causing different nucleotides to be attached.

This method is being used by the Illumina GA II (Illumina Incorporated, CA, USA) with solid-phase amplification and by Applied Biosystems SOLiD (Life Technologies Corporation, CA, USA) with emPCR.

Single-molecule templates

In this method, no amplification is required. There are three different methods used with single-molecule templates. In first method, the primers are attached to a

solid-surface. Then by adding adapters the fragments are hybridized to the attached primers. In second method, the fragments are attached to the surface by priming and extending single-stranded, single-molecule templates from immobilized primers. In the third method, an enzyme instead of a primer or fragment is attached to a solid surface.

This method is being used by Helicos HeliScope (Helicos BioSciences Corporation, MA, USA).

1.1.2 Sequencing and Detection

The sequencing techniques are linked with detection and imaging of the result. There are several methods used in sequencing.

Cyclic reversible termination (CRT)

This method works by adding nucleotides that terminate the DNA synthesis. In each cycle a single nucleotide is added, then an image taken and finally termination is reversed. The nucleotides are modified with reversible termination and labeled with fluorescent label. There are several ways how the termination is done with each manufracturer using their own system. This method also requires a modified DNA polymerase which has spurred development and solutions have been suggested to overcome this.

CRT uses either one or four colour labels which affects the image analysis. With the four colour system, the image shows which positions added a label. With one colour system, the image shows if a given position added a colour or not.

This method is being used by the Illumina GA II with four colour labels.

Sequencing by ligation

The method is run in two cycles: ligation cycle and primer reset cycle. First an universal primer is hybridized to the library fragment as part of primer cycle. In each ligation cycle, a set of four fluorecently labeled di-base probes are allowed to hybridize to the fragment. Once the probes are hybridized, the universal primer and probe next to it are ligated together by DNA ligase. After detection the fluorescent label is cleaved off and the next ligation cycle is commenced. After all the ligation cycles are done the primer is reset and new ligation cycles are done. Each ligation cycle detects every 5th base of the fragment and as the primer reset moves the primer by one base, 5 primer reset cycles are needed to cover the all bases the produced read.

This method is being used by the Applied Biosystems SOLiD.

Single-nucleotide addition

This is also called Pyrosequencing. In this method, nucleotides are added one at a time with DNA polymerase, but no modified nucleotides are involved. In addition, no fluorescent markers is used; instead the release of phosphorus in nucleotide polymerisation is converted into light by a series of enzymatic reactions. This requires the intensity of the light to be measured. The intensity of the signal, measured after each added nucleotide, shows if and how many of that particular nucleotide is attached.

This method is being used by the Roche 454 (454 Life Sciences, a Roche company, CT, USA).

Real time sequencing

This is thought to be the next new method with sequencing. The Pacbio RS sequencer (Pacific BioSciences, CA, USA), which should be brought to market 2011, uses real time sequencing. In their method, an enzyme is attached to a surface and the nucleotides are marked with fluorescent markers. Colour is emitted when it's removed from the nucleotide as it's attached to the DNA strand by the polymerase enzyme. Result is a graph with intensity of four colours versus time.

1.1.3 Data analysis

A single sequencing run produces two different kinds of data. First is the raw result comprising of the images or intensities. Usually there is also a software package provided by the manufracturer which converts the images to sequence fragments. The amount of the reads and their length depends on the platform and run parameters.

Due to very large amount of data produced by the next-generation sequencers the handling of results required considerable development over the tools used in first generation sequencing. In most cases the tools are not memory efficient or fast enough to handle millions or billions of short reads. For example, classic sequence alignment tool such as blast could take a week to make alignments of short reads against the reference sequence. In many cases, completely new algorithms have been taken into use to overcome these problems.

In addition to the classical *de novo* sequence assembly, sequencing is nowdays used for many other applications. For example, microarray-based technologies to measure relative amounts of transcripts are being replaced with RNA sequencing. Chromatin immunoprecipition studies and genome-wide profiling of epigenetic markers are also being done with the new sequencers.

Figure 1.1: Picture of the Illumina Genome Analyzer IIx with paired-end module (in left) in FIMM.

1.2 Illumina Genome Analyzer II

The Illumina Genome Analyzer II came to market in 2008. The current platform is called GA IIx. The Illumina has also put out the more advanced version of the GA IIx which is based on the same chemistry: HiSeq 1000 and HiSeq 2000. Both achieve almost 20 fold increase in throughput. The latest addition to the Illumina sequencing platforms is MiSeq which is smallest of the instruments with smallest throughput with about 1Gb per day.

In addition to the sequencing machine, there is usually a cluster station and a paired-end module. The cluster station is needed for the sequencing but one such a station can serve more than one sequencer. Cluster station is used to generate clusters as described in section 1.2.2 on page 6. The paired-end module is needed for being able to read the sequence of a short DNA fragment from both ends of the molecule. On average a paired-end run of 100 + 100 bps with the Illumina GA II will take a week or little over, depending on the specifics of the run.

The operational workflow for Illumina GA II has four steps

1. Library preparation
2. Cluster generation

3. Sequencing
4. Data analysis

1.2.1 Library preparation for DNA sequencing

The DNA used in sequencing can be whole genome or targeted areas of interest. There should be 0.1 - 3 μg of DNA depending of the application. First the DNA to be sequenced is fragmented (Figure 1.2 A). After fragmentation, the ends are repaired and a nucleotide A is added as an overhang (Figure 1.2 B) to facilitate the adapter ligation. Adapters are then ligated to the overhang and the fragments are amplified by PCR (Figure 1.2 C). Adapters allow the fragments to be amplified with predefined set of PCR primers.

This takes about 6 hours of which 3 hours is hands-on.

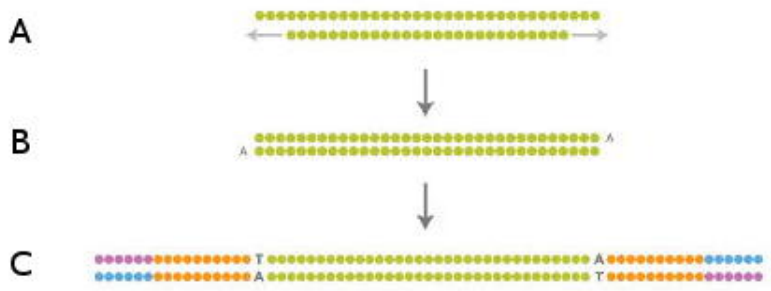

Figure 1.2: Solexa GA II step 1, library preparation (Illumina, GAB, 2010).

1.2.2 Cluster generation

The DNA fragments are first attached or hybridized to a silica slide referred as a flow cell (Figure 1.3 A). There are 8 separate lanes in a flow cell. This allows 8 separate sequencing events in one run. Each fragment library goes into one lane. Usually this means that 8 samples are analysed in one run. However multiplexing kits from Illumina allows up to 12 samples in one lane with total of 92 samples per flow cell. In Multiplexing, an index tag (a small sequence) is attached to each sequence fragment which varies between samples. This tag then allows the produced sequences to be separated per sample. Multiplexing is useful with smaller targets within large genome or smaller genomes, such as bacteria.

The next step after hybridization is bridge amplification. In bridge amplification, the second strand is synthethised to the attached fragment using the adapter as primer. The strands are then separated with denaturation and bound to the adjacent primer creating a bridge (Figure 1.3 B). This process is repeated 35 times to create

a clusters of clonally amplified DNA molecules also known as sequencing templates (Figure 1.3 C).

Lastly the reverse strands are removed and the sequencing primers annealed to the clusters (Figure 1.3 D).

This in total takes about 4 hours with less than 10 minutes hands-on.

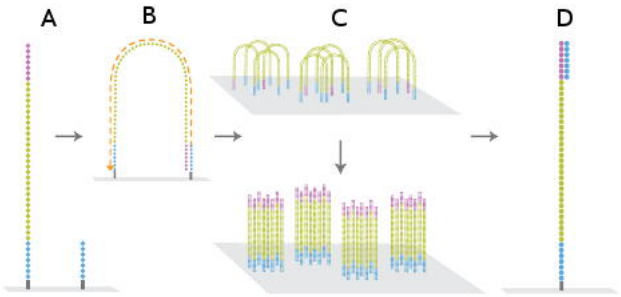

Figure 1.3: Solexa GA II step 2, cluster creation (Illumina, GAB, 2010).

1.2.3 Sequencing

The sequencing can be divided into two parts, the sequencing and base calling. Sequencing is carried out in three steps. First, a modified nucleotide with a fluorescent label is added to the primer according to the template fragment. Then an image is taken, and finally the termination group and fluoropore is cleaved from the added nucleotide (Figure 1.4 A). This is then repeated to create reads of the wanted length (Figure 1.4 B). Current chemistry supports up to 150 nucleotide reads. Finally, the bases are called with computer software (Figure 1.4 C).

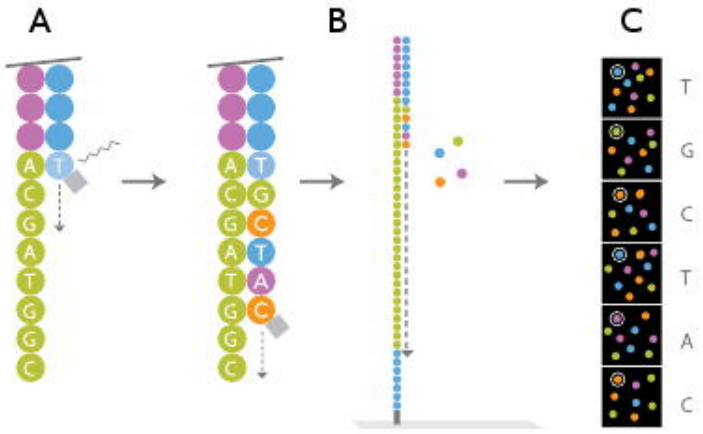

Figure 1.4: Solexa GA II step 3, sequencing (Illumina, GAB, 2010).

In paired end sequencing the second read is done after the first strand is completely sequenced. First, the newly made strand is removed with denaturation. The clusters are then redone in a similar fashion as in bridge amplification. The difference is that the forward strands are then removed instead of the reverse ones (Janitz, 2008). The sequencing of the second strand is done as described above.

This takes from 2 to 7 days (depending on the length of the reads) for single end runs and 5 to 14 days for paired-end runs with about 30 minutes hands-on time.

1.2.4 Data analysis

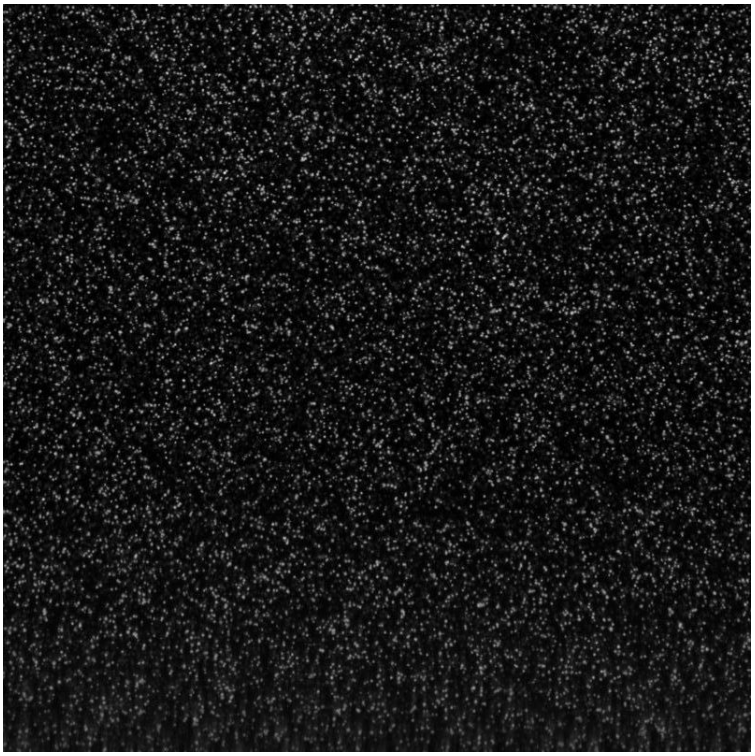

Figure 1.5: GA II image. A section of an image from the Illumina GA II for one nucleotide in sequencing. Each white dot represents a cluster.

The Illumina GAII generates images as the raw data. These images are tiff (Adobe, TIFF v6.0 specification, 2010) files and there is one file for each nucleotide for each cycle. An example of the image that is created is shown in figure 1.5. These images are automatically converted into sequences using Illuminas real time analysis software. This produces the sequences which are then converted into fastq file format (Cock et al., 2009). For single-end sequencing runs there is one file and for paired-end runs there are two.

In an average run there are between 2 to 16 Gb (in hard drive space) and from 2 to 8 Gb (in base pairs) of sequences are produced per lane with each having from 40 to 120 million reads. The numbers vary very much due to cluster density, read length and whether paired-end sequencing is used.

1.3 Whole Genome Exome Kits

With current NGS platforms, sequencing of the whole genome or targeted parts of it has been enabled. One of the latest developments are the exome kits. These allow targeted capture and sequencing of the whole human exome. Currently there are five kits for sequencing human exomes, two produced by Nimblegen (Roche Nimblegen Inc, WI, USA), two by Agilent (Agilent Technologies Inc, CA, USA) and one by Illumina. In this experiment the first kits by Nimblegen and Agilent are used.

Definition of the whole exome however is not entirely clear. Exome should be all exons in the genes of the organism, in this case human. Not all genes locations are known however so different databases such as EnsEMBL, NCBI RefSeq and UCSC show a bit different set of genes. Creating an exome from these would have differences. Both exome kits are designed against Consensus Coding Sequence (CCDS) (Pruitt et al., 2009) for human genome version 18 (hg18).

1.3.1 Nimblegen SeqCap EZ Exome Library

The Nimblegen SeqCap EZ kit has 2.1 million DNA probes with length of 60 to 90 basepairs. On average over 10 probes per exon in the human genome, to capture DNA fragments representing the exons by hybridization. The SeqCap Ez is designed to be simple to use, as all probes are in one tube. It also includes some built-in controls to evaluate the quality of the capture and sequencing in the lab (Nimblegen, Exome brochure, 2010). The SeqCap Ez needs from 1 to 5 μg of input library. Performance of the kit is shown by nimblegen in table 1.1.

1.3.2 Agilent SureSelect Human All Exon Kit

The Agilent SureSelect kit is aimed to cover the entire human exome (Agilent, Exome brochure, 2010). The speficifications state that it covers 38Mb or 1.22% of human genome corresponding to all consensus coding sequence regions (Pruitt et al., 2009). Agilent uses 120 basepair long RNA probes. This kit is also packed into a single tube like Nimblegens SeqCap Ez. However it is noted that only 500 ng of prepared sequence library or from 1 to 3μg of genomic DNA is required. Several alternative

sizes of the kit are available, from 5 to 5000 reactions. Performance of the kit is shown by agilent in figure 1.6.

1.4 Other NGS Pipelines

Currently there are not that many pipelines available in the literature or otherwise. In some cases it is most likely considered to be part of infrastructure, and thus not published or put on websites. For example, the Sanger Institute has its own similar pipeline, but nothing can be found on their website.

A Genome Analysis Toolkit known as GATK (Genome Analysis Toolkit, 2010) has been developed by the Broad Institute in the USA. GATK is a software library for developing tools to make analysis of NGS data simpler. However, since it is not a pipeline nor a tool as such, it is not being used in the produced pipeline. Some of steps, such as variant and indel calling, in the pipeline described in this thesis could alternatively be conducted using GATK.

There is a one published pipeline for NGS from 2009 (Jex et al., 2010). However, this pipeline is aimed at mitochondrial DNA and assumes certain methods in sequencing. Thus, it is not really usable for our needs.

A pipeline called inGAP also exists (Qi et al., 2010). It uses similar tools as this pipeline. However, inGAP is missing some of the functionalities such as statistics and paired-end anomaly detection which are available in our pipeline. inGAP has some extra flexibility, such as supporting single reads.

There are several commercial software packages which are able to perform variant calling, such as NextGENe (NextGENe, 2010) and Alpheus (Alpheus, 2010). However, there might be a need for the user to run the pipeline with different options, or hardware configurations could change. In many cases licensing restrictions may exist in proprietary software packages that can easily hamper the practical use. For this reason, while proprietary software packages can be useful, they do not remove the need for this pipeline.

Table 1.1: Performance of Seq Cap Ez as provided by Nimblegen (Nimblegen, Exome brochure, 2010).

Sequencing Output	1 Lane	2 Lanes	4 Lanes
Total Sequence (Gb)	1.2	2.8	5.5
Total Number of Reads (million)*	15.2	35.5	69.2
Percentage Read on Target	60.8%	60.2%	61.9%
Percentage Target Base Covered by 1+	96.4%	97.5%	98.2%
Percentage Target Base Covered by 5+	86.0%	92.9%	96.3%
Percentage Target Base Covered by 10+	70.7%	86.3%	93.6%
Detection Rate for Known Heterozygous SNPs in Exon Targets (6318)	88.8%	95.4%	97.9%
Detection Rate for Known Homozygous SNPs in Exon Targets (4787)	94.3%	96.3%	97.3%

* This is total number of raw sequence reads (paired-end 2x40 base reads). The reads are filtered for uniquely mapped reads for downstream SNP analysis.

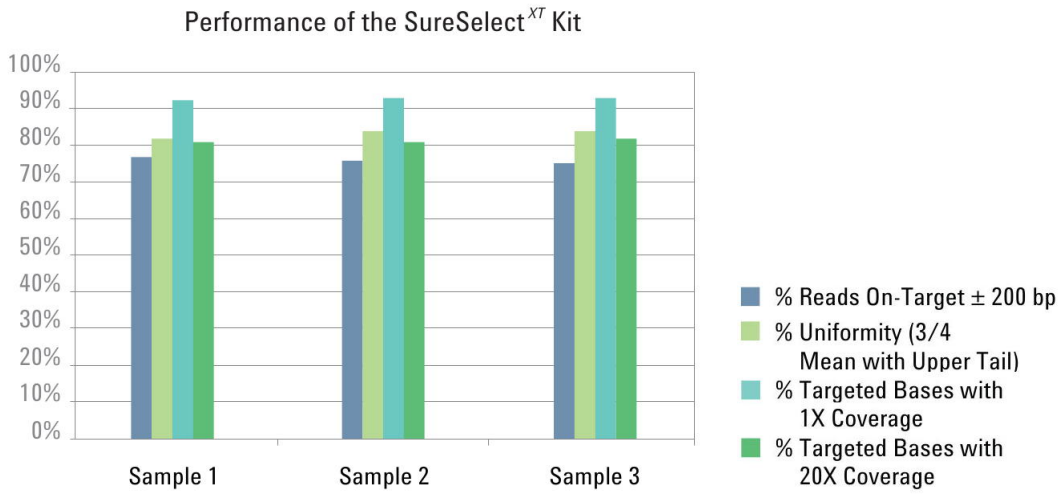

Figure 1.6: Performance of SureSelect capture as provided by Agilent (Agilent, brochure, 2011). The data shown has been done using SureSelect human X chromosome baits and not all exon capture.

Chapter 2

Aims

The thesis will have two aims. Developing of a functional pipeline for variant calling is the first aim. Second aim is to use the pipeline for comparing two Exome sequencing kits, available from two manufacturers.

2.1 The pipeline

The developed pipeline aims to 1) provide basic information about the quality of experiment, 2) do the alignment of the sequencing products to the reference, 3) identify SNP and copy-number variants (CNV), 4) annotate known and novel variants and 5) identify anomalies.

2.2 Exome kit comparison

The aim in exome comparison is to see how well the kits perform in sequencing the human exome. Coordinates provided by both manufacturers were compared to the reference exome (Pruitt et al., 2009). Sequencing results of the libraries created from the two sequence capture kits were evaluated by several means such as coverge through the exome with several target sets as well as whole transcripts or genes, ability to indentify known and novel variants and how much variability the pairing showed.

Chapter 3

Materials and Methods

This chapter shows the detailed explanation of the pipeline itself. The pipeline has two parts which are executed separately: indexing and variant calling pipeline (VCP). The indexing is a separate section where pre-processing is done based on the reference sequence. This is required before the pipeline can be executed as various steps, most notably alignment, use the indexes created. In the thesis the different parts of the VCP are described in different sections.

Different sections of the pipeline are shown in the graphs of this chapter. In those graphs the boxes represent data while the ellipses represent a program to be run. Note that not all the files created are represented in graphs, but only the ones which are considered to be important. Arrows are used to connect programs showing that first programs output is the second ones input.

Most of the programs used here are publicly available. In addition, several scripts have been developed to make the pipeline run automatically. Most of these have been written with perl scripting language, but some are done with R and bash.

The original plan of the pipeline was designed by Pekka Ellonen from FIMM Technology Centre.

3.1 The pipeline overall

The pipeline sections and their main inputs and outputs are shown in figure 3.1.

The workflow is currently executed with a python script. The script is able to restart the workflow from the middle of the workflow if something fails or if an option is modified and workflow rerun. At the start of the pipeline, the versions of external programs used are printed out. During the run the pipeline does some clean up to the files produced. Larger of the intermediate files are removed, except the

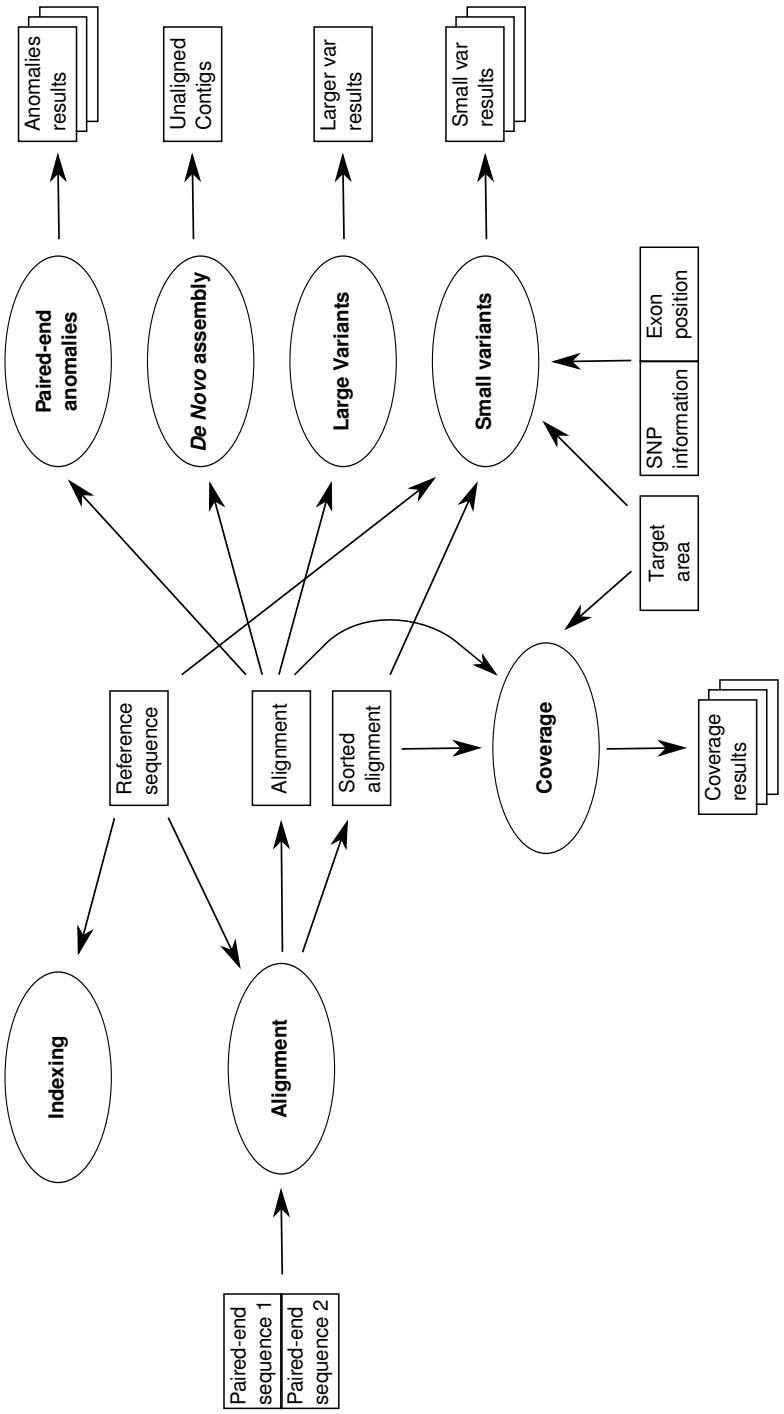

Figure 3.1: VCPipeline sections and their interactions as well as overall inputs and output of the pipeline. The ellipses describe the sections of the VCP and boxes are files. Stacked boxes represent several separate types of data. These stacked boxes are only used in results to to emphasise that there is multiple results from the section. The index section is required in order for Alignment, Small Variant and Coverage sections to run.

alignment results which are saved. Some of the larger result files are also compressed using `gzip` program.

For indexing, the inputs are given on command line as the indexing only needs the reference sequence file. For the VCP, the inputs to are given with a configuration file. The configuration file has three sections. First section lists all the input files used by the pipeline as well as output directory. Second section lists selected options for tools within the VCP. Third section has limits and options which affect the pipeline instead of an individual program.

The configuration file for the pipeline can be seen in appendix A on page 63.

Pipeline inputs

The main input files for the VCPipeline are: the reference sequence and the two paired-end reads. In addition, the VCPipeline can use a target area, a SNP information and an exon position files, but these are not required.

The reference sequence should be in fasta file format. Fasta format is simple format in which the sequence entry has a header line starting with a `>` character. After header it has one or more lines with the sequence. The suggestion is to have up to 60 characters of sequence in one line. Multiple sequence entries can be in a single file. However, the indexing for alignment uses only part of the name, stopping at the first space. This part before the space must be unique so that the index created is valid. The part will be used as the reference item name of the sequence entry in the pipeline. The resulting item name should be same as what is used in possible additional inputs in order for them to match the results of the pipeline. The order of the items in the reference sequence file defines some of the sorting at some points later on, for example order of reference items in anomaly plots.

```
@HWI-EAS418_0100:1:1:1:1553#0/1
NGACTCCAAGCAAACTCCACATGCAGGCAATGGAACTTAACANNNCNNNCAGCGGGAAG
+HWI-EAS418_0100:1:1:1:1553#0/1
DMVUSXVRRUTNTPPPXWQSROWWRBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBB
@HWI-EAS418_0100:1:1:1:282#0/1
NGTTGAAAGCTATGGTACAATATGCAAAGTCTCTTAGTGAAGNNNTNNNAAAATCAACA
+HWI-EAS418_0100:1:1:1:282#0/1
DN[[Y[[ZYY[SWZWUYY[[[[[[[VUVVVY[ZBBBBBBBBBBBBBBBBBBBBBBBBB
@HWI-EAS418_0100:1:1:1:720#0/1
NTGACGTTCTACACTTCCAGTGTCTGATCCGGAGTGTTGCTCNNNTNNNTCATCCCCAC
+HWI-EAS418_0100:1:1:1:720#0/1
DPUUXVWWXTPUUWXWXXWWWXXWWWSWWWUTBBBBBBBBBBBBBBBBBBBBBBBBB
```

Figure 3.2: An example from the start of a fastq file produced by Illumina software from the GA II paired-end sequencing. This example is from the Agilent exome comparison data.

The paired-end read files should be in fastq file format (Cock et al., 2009). There should be two files with one file having one end of the pairs and the other having the other end of the pairs. An example of start of a fastq file can be seen in figure 3.2.

Target area should be gff file format (GFF specification, 2010). This is assumed to be target of the sequencing and there will be additional information regarding this area in the coverage section of the pipeline. The track names given are not checked so this file can only have the areas which are relevant to the target. An example of the file can be seen in table 3.1. The file should be sorted by the position (first by start point, then by end point) in the reference entry so each reference entry is in its own section. The target areas can overlap, but a target area cannot be within another target area.

The SNP information file is a tab separated file without a header row. This file should have one sequence variant in each row. The file should be sorted by the position in the reference entry in a similar fashion as the target area file above. The same position in the reference can have multiple entries and names of the SNPs do not have to be unique. An example of the SNP information file can be seen in table 3.2.

The exon position file is also a tab separated file without a header row. This file has position of all exons in the reference genome and gene, transcript and protein information related to the exon. It should be sorted by the position in reference entry in a similar fashion as the target area file above. There is no unique restrictions outside of the ordering. An example of the exon position file can be seen in table 3.3.

3.2 Indexing

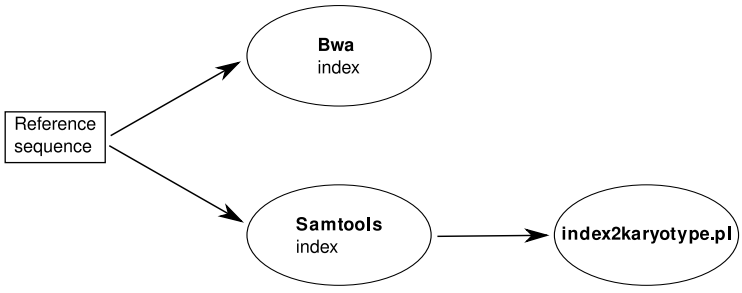

Figure 3.3: Steps of index creation for the pipeline.

The indexing is a separate part of the entire pipeline and its workflow is shown in the figure 3.3. It must be run every time the reference sequence changes and must

Table 3.1: The target area file
This is in gff file format (GFF specification, 2010). The data is from CCDS (Pruitt et al., 2009) target areas used in exome comparison.

1	bed2gff	feature133814	69091	70008	0	+	.	group133814;
1	bed2gff	feature133815	367659	368597	0	+	.	group133815;
1	bed2gff	feature133816	621096	622034	0	+	.	group133816;
1	bed2gff	feature133817	861322	861393	0	+	.	group133817;
1	bed2gff	feature133818	865535	865716	0	+	.	group133818;
1	bed2gff	feature133819	866419	866469	0	+	.	group133819;
1	bed2gff	feature133820	871152	871276	0	+	.	group133820;
1	bed2gff	feature133821	874420	874509	0	+	.	group133821;
1	bed2gff	feature133822	874655	874840	0	+	.	group133822;

Table 3.2: The SNP information file.
This file can have other similar entries as well and in the exome comparison there is also CNVs in the file.
First column has the reference entry name (in this case the chromosome number). Second column has the name of the variant. Third and fourth columns have the start and end of the variant respectively. If the end is smaller than start then there is an insert between the two positions. Fourth column has alleles of the variant. Fifth column has the consequence of the variant.
This data is from the Ensembl (Hubbard et al., 2009) version 57 with variants version 37b.

1	rs55998931	10492	10492	C/T	UPSTREAM
1	rs62636508	10519	10519	G/C	UPSTREAM
1	rs10218492	10828	10828	G/A	UPSTREAM
1	rs10218493	10904	10904	G/A	UPSTREAM
1	rs10218527	10927	10927	A/G	UPSTREAM
1	rs28853987	10938	10938	G/A	UPSTREAM

Table 3.3: The exon position file.
First column has the reference entry name (in this case the chromosome number). Second and third columns have the start and the end of the exon respectively. Fourth column has the gene in which the exon belongs to. Fifth column has the transcript in which the exon exists in and sixth the protein of that trascript. Fifth and sixth columns can be empty. The same position can have multiple entries allowing the position to be in multiple genes, transcripts or proteins.
This data is from the Ensembl (Hubbard et al., 2009) version 57.

1	11874	12227	ENSG00000223972 (DDX11L10)	ENST00000456328	ENSP00000410013
1	12010	12057	ENSG00000223972 (DDX11L10)	ENST00000450305	
1	12179	12227	ENSG00000223972 (DDX11L10)	ENST00000450305	
1	12613	12697	ENSG00000223972 (DDX11L10)	ENST00000450305	
1	12613	12721	ENSG00000223972 (DDX11L10)	ENST00000456328	ENSP00000410013
1	12975	13052	ENSG00000223972 (DDX11L10)	ENST00000450305	
1	13221	13374	ENSG00000223972 (DDX11L10)	ENST00000450305	
1	13221	14412	ENSG00000223972 (DDX11L10)	ENST00000456328	ENSP00000410013

be done before running the VCPipeline itself. Indexing consists of three tools: bwa (Li and Durbin, 2009), samtools (Li et al., 2009) and index2karyotype.pl script.

The bwa is the alignment tool that is used in the VCPipeline. It needs an index to the reference sequence and `bwa index` command creates that. In small variants sections a second index is needed for the samtools tool package. `samtools index` is a command which creates the index file required. Lastly the `index2karyotype.pl` is executed. It is a script which was created to convert the samtools result for another tool used in paired-end anomalies part of the pipeline. It takes in the samtools' index file and reformats it.

Reference sequence is a fasta file which can be have multiple sequence entries. However one reference sequence cannot be in multiple files.

3.3 VCPipeline - alignment

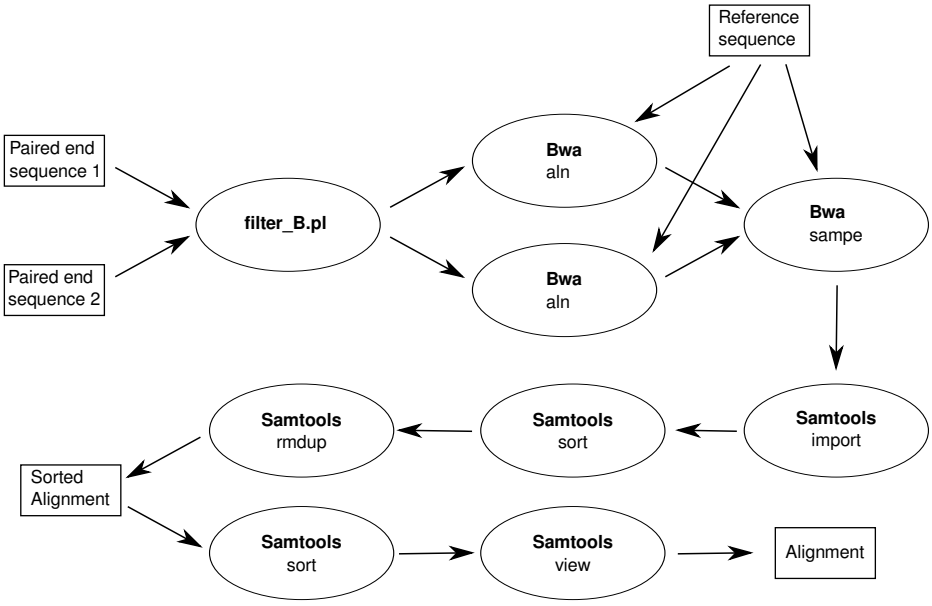

Figure 3.4: Steps to align the reads to reference in the pipeline.

First step is to do a filtering on quality score B. The Illumina documentation mentions that with Casava 1.5 onwards the ends of the reads may have a block B quality scores and that these block should not be used in further analysis. The `filter_B.pl` script trims the sequences and removes pairs from which one or both reads end up shorter than the defined minimum (by default 36). Then the actual alignment to the reference is done and for this `bwa` is used. The workflow is in figure 3.4. The alignment is done in 3 steps. First, the individual reads are aligned to the reference with `bwa aln` command. The command writes the result in to a

binary format which marks the coordinates within the reference. Then the individual reads are paired and written out in sam (Li et al., 2009) formatted alignments file with `bwa sampe` command. The result of this is the raw alignment.

Next, the raw alignment is cleaned. First, the raw alignment file is converted to binary sam (bam) format with `samtools import` command. Then the bam file is sorted according to the positions of the alignments in the reference with `samtools sort`. The sorting is required for the `samtools rmdup` command which is executed next. This removes the possible PCR duplicates created during the sample preparation. The result of this is the sorted alignment binary file which is used as input in the small variants section later on.

Last step is to get a name sorted sam file for other sections. To get this the position sorted binary bam file is re-sorted with `samtools sort`. This time the file is sorted by the name of the read, creating a bam file with read pairs next to each other. Finally, the file is then converted to sam format with `samtools view` command.

The alignment tool could be changed with sufficient ease if that was wanted as long as the result of the alignment is in sam file format. Such a change would require redoing the first part of this section of the pipeline as well as part of indexing. Switching to an aligner which does not output the sam format would require converting the alignment result into sam format so that the samtools and other scripts would be able to use the alignment result.

3.4 VCPipeline - small variants

Small variants section is for small indels (only few nucleotides) and SNPs. The workflow for that can be found in figure 3.5. This section produces the most of the main results of the pipeline. The main inputs are: the reference sequence, the position sorted binary alignments, the target area, the SNP information table and the exon position table. Two separate tables are created as result: genotype and variant. The genotype table has all the previously known SNPs in target area which were sequenced, while variant table lists all SNPs found in sequencing above certain quality thresholds. A quality graph is also produced for the SNPs and variants. This part uses samtools and scripts done for the pipeline.

First for variants, `samtools pileup` is called to retrieve all differences from the alignment to the reference. This raw result is then filtered using `samtools.pl varFilter` and selections based on the quality values. The result is then reformatted into a table as a result for the user as well as to draw graphs from. This conversion is done with `pileup2graph.pl`. This script reformats the pileup commands output,

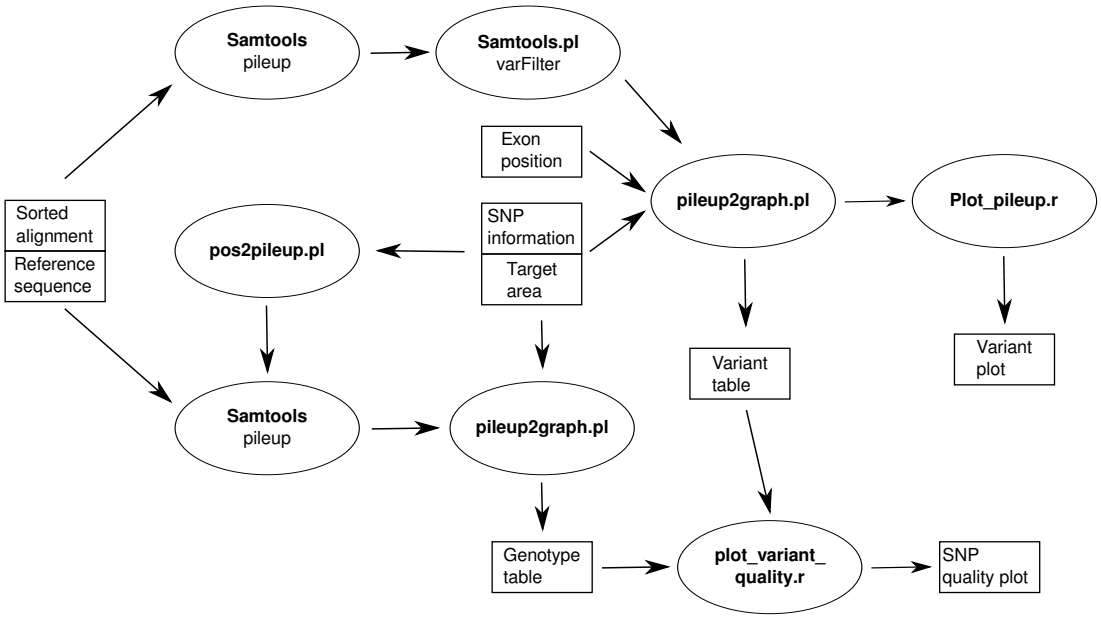

Figure 3.5: Steps of the pipeline to find small variants.

and allows additional filtering to be done based on read depth and quality ratio from the sum of quality between variant and reference bases (by default depth requirement is 10 and ratio should be 0.8 where 1 is reference).

Finally for the variants, the `plot_pileup.r` is executed. It creates a pdf with a graph showing call depth, call ratio and quality ratio for each entry in variant table. The call depth is the amount of reads covering the entry. Quality ratio is the same as described above and call ratio is the ratio between variant and reference calls. This result can be used for a simple overview for assessing the quality of the SNPs.

Then the genotype table is created. It uses the SNP information table to find all positions of SNPs in the target area of the reference with a `pos2pileup.pl` script. The result is then given to `samtools pileup` which retrieves those positions from the alignment regardless of whether it differs from the reference or not. The results are then converted to a table with `pileup2graph.pl` as before, but this time without any filtering.

Lastly a quality graph of the variants as well genotypes are done. The file is produced from variant and genotype tables producing a file with two plots using `plot_variant_quality.r` script.

If the SNP information or target area positions are not given by the user, the genotype part is not executed. The quality graph is done but with only the variant plot.

3.5 VCPipeline - coverage

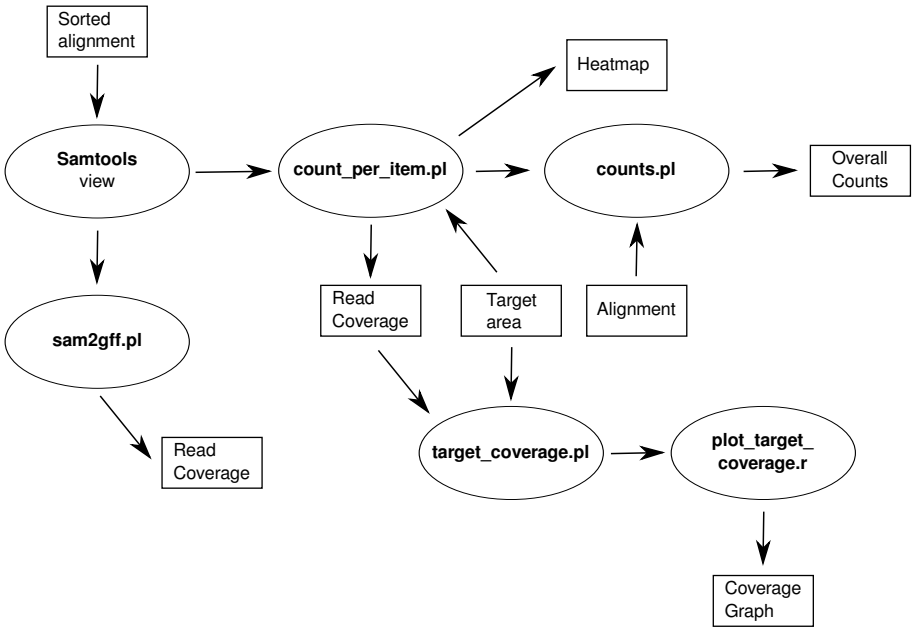

Figure 3.6: Steps of the pipeline to get coverage and statistics from reads alignments.

The coverage section counts some basic statistics from the alignment. In addition, it produces two files which have the coverage of reads and graph showing the quality of the coverage in target areas. It uses samtools to get the data and the scripts done for the pipeline. Workflow is seen in figure 3.6.

First, the reads which were aligned to the reference are separated out. This is done with `samtools view` command. It reads in the position sorted binary alignment file and prints out in sam format only the reads which were aligned against the reference. From this, two different things are done: coverages of reads and targets and statistics counting.

The counts start with the `counts_per_item.pl` script. It prints out the count statistics per sequence item in the reference. In addition, it produces a coverage file and a heatmap table which has the amount reads in any individuals target area. The script uses an index file created earlier in indexing phase and takes in a minimum fold coverage value, meaning that at least that many reads should cover a nucleotide for it to be considered (by default 5). The tool counts both one fold coverage and the user given fold coverage. Then `counts.pl` combines these results and prints out overall statistics.

The read coverages are printed in two separate formats. First format is done with a `sam2gff.pl` script. It converts the reads into a static window size (by default 50)

and counts how many times that is covered by reads. The format of the result is gff. Second is done with `counts_per_item.pl` script which was explained above. The resulting file is in bedgraph (BedGraph Track Format, 2010) format which has coverage for all positions. The bedgraph coverage file is then compared to the target areas with `target_coverage.pl`. This script groups the windows with similar score and prints out the counts of the windows in the groups. The result is a table from which `plot_target_coverage.r` script draws the graph which shows the overall target area coverage.

3.6 VCPipeline - larger variants

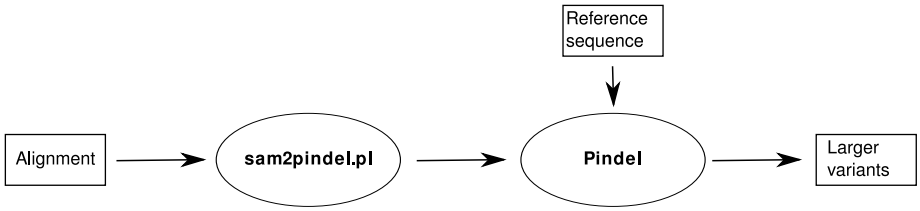

Figure 3.7: Steps of the pipeline to find larger variants.

In order to find the larger variants, a program called pindel (Ye et al., 2009) is used. Workflow is shown in the figure 3.7.

Pindel uses as an input those read sequences of which the mate is aligned to the reference and the read itself is not. In order to get these reads, a script `sam2pindel.pl` is executed. It goes through the alignment result and prints out the reads described above. Then the `pindel` is run. It gives out three files which determine different sorts of larger variants: deletions, insertions and special deletions with non-template insertion around break points.

3.7 VCPipeline - *de novo* assembly

De novo assembly is done with velvet (Zerbino and Birney, 2008). The aim however is not to do a full assembly of all the reads. Instead the reads which were not aligned to the reference are used here. Any of the assembled contigs could then be used to search with blast or similar algorithm from a general sequence database to see if something interesting could be found. Note that this section is only run if there is less unaligned reads than the number given by user.

First, the reads which were not aligned to the reference are taken from the alignment with `sam2velvet.pl` and printed out as set of fasta format sequences. This set of reads is then pre-processed with `velveth` for assembly. Finally, the

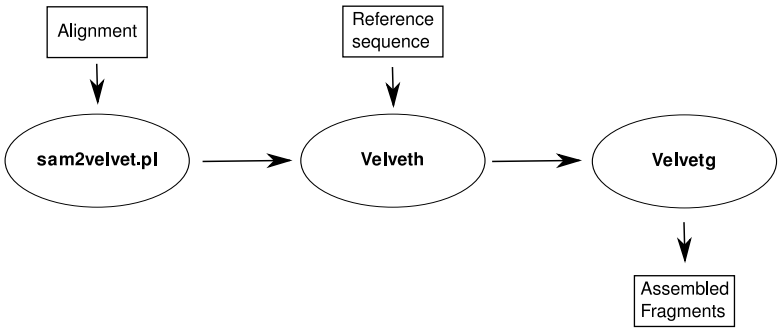

Figure 3.8: Steps of the pipeline to do *de novo* assembly.

contigs are assembled with `velvetg`. This takes in the median length between the reads and minimum contig size given by the user. The result is a set of assembled contigs of given minimum length.

3.8 VCPipeline - paired-end anomalies

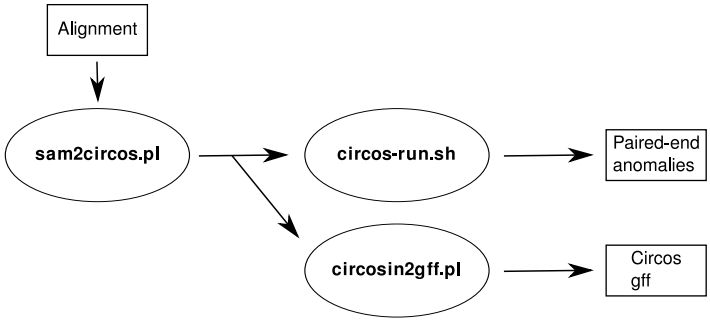

Figure 3.9: Steps of the pipeline to find paired-end anomalies.

The paired-end anomalies are visualised with circos (Krzywinski et al., 2009). The aim in this section is to find larger groups of sequence pairs which are unusually paired. This means that they are too far apart in same chromosome or even in different chromosomes.

First, the pairs which were aligned anomalously are retrieved from alignment with `sam2circos.pl`. This script finds all pairs which have ends in different sequence entries in reference or farther than given value (default 3000) nucleotides apart in same sequence entry. There is an option which defines the minimum amount of overlap in order to draw that pair (by default 3). The resulting pairs are written in given directory with a file for each sequence entry in the format required by `circos`. Each file has the pairs of which either end of the pair is in that reference entry and there is at least minimum amount of overlap as defined earlier. Next, each of

the sequence entry files are converted into gff files with `circosin2gff.pl`. The gff tracks created are not sorted.

Finally, the `circos-run.sh` is executed. The script runs `circos` for each sequence entry which creates a picture from that entry. The result shows the approximate position in items where the overlapping reads start and end. The thickness of the line depends on how many reads overlap. The result for an entry has all pairs which have one end on that item so each pair is showed twice from both ends if it goes from one item to other.

3.9 Exome kits

The modifications to the manufacturers' recommended protocols were done in order to allow additional QC measures between the steps and to remove unnecessary differences between the protocols.

3.9.1 Nimblegen SeqCap EZ

The samples were prepared mostly according to the SeqCap EZ protocol (Nimblegen, Exome protocol, 2010) and illumina paired-end protocol (Illumina, PE sample prep, 2010), which is referred by the SeqCap EZ protocol. The modifications to the protocol are listed below.

Sample preparation, Illumina guide

The DNA was fragmented using Covaris S2 (Covaris, MA, USA) with Covaris microTubes with AFA fiber. The following setting were used in fragmentation: duty cycle 10%, intensity 5, cycles per burst 200, time 180 seconds, set mode frequency sweeping.

The sample preparation kit was changed from Illumina Paired-End Genomic DNA Sample Prep Kit to NEBNext^{TM}DNA Sample Prep Master Mix Set 1 (New England Biolabs Inc, MA, USA). This would affect following sections in illumina paired-end protocol: perform end repair (page 16), adenylate 3' ends (page 18) and ligate adapters (page 19).

The quality, quantity and the size of the sample libraries were verified with Agilent Bioanalyzer DNA1000 kit (Agilent Technologies Inc, CA, USA) after steps: perform end repair (page 16) and ligate adapters (page 19).

In ligation, 25 pmol of adapters were used. 200 - 300 base pairs were cut from gel.

SeqCap Ez guide

Chapter 4. sample library amplification using LM-PCR. 24 pmol of both primers and 20 ng of sample DNA were used in the PCR reaction. The PCR was run total of 8 cycles with 4 reactions per sample.

Chapter 7. captured DNA amplification using LM-PCR. 100 pmol of both primers and 4,5 pg of sample DNA were used in the PCR reaction. The PCR was run total of 14 cycles with 6 reactions per sample.

After capture the concentration was measured with a qPCR. A 2-fold dilution series of the standard was used to obtain a standard curve ranging from 50 pM to 1,56 pM. The qPCR was performed with Finnzymes DyNAmo SYBR Green master mix (Finnzymes Oy, Espoo, Finland) using 10 pmol of primers.

Chapter 8. Measurement of Enrichment Using qPCR. Following areas were used as targets in this: NSC-0237, USF1_exon, NSC-0272, JMY_exon, Chr16_cons. The primers for these areas can be found from appendix B.

3.9.2 Agilent SureSelect Human All Exon Kit

The samples were prepared mostly as the SureSelect protocol specifies (Agilent, Exome protocol, 2010). The modifications to the protocol are listed below.

Sample preparation

The NEBNext^{TM}DNA Sample Prep Master Mix Set 1 sample preparation kit was used instead of Illumina Paired-End Genomic DNA Sample Prep Kit. This would have an effect in Steps 4 - Repair the ends, 6 - Add 'A' Bases to the 3' end of the DNA fragments and 8 - Ligate the paired-end adapter.

Step 1, Shear DNA, page 16. The shearing was done with Covaris S2. The settings differ from of the protocol with time set to 180 seconds.

Step 8, Ligate the paired-end adapter, page 23. 25 pmol of adapters were used in the ligation.

Step 11, Amplify adapter-ligated library, page 27. 24 pmol of both primers and 20 ng of sample DNA were used in the PCR reaction. Also, the PCR was run total of 8 cycles with 4 reactions per sample.

Step 13, Assess quality and quantity with Agilent 2100 Bioanalyzer, page 29. The concentration was measured with Bioanalyzer and based on the analysis, 500 ng of sample was used for capture.

Post-Hybridization Amplification and Cluster Station Preparation

After the capture the concentration was measured with a qPCR. A 2-fold dilution series of the standard was used to obtain a standard curve ranging from 50 pM to 1,56 pM. The qPCR was done with Finnzymes DyNAmo SYBR Green master mix using 10 pmol of primers.

Step 1, Amplify the captured library, page 42. Based on the qPCR result above, the PCR used 11,5 pg sample per reaction and 50 pmol of primermix. PCR was then run 16 cycles and 6 reactions per sample.

The result of the PCR was then verified with enrichment qPCR. Following were used as targets in this: NSC-0237, USF1_exon, NSC-0272, JMY_exon, Chr16_cons. The primers for these areas can be found from appendix B.

3.9.3 Illumina Human 660 Quad chip

The quality of the variant pipelines variant calling was evaluated using a Illumina Human 660 Quad SNP-chip (Illumina SNP chip). The sample used in the exome comparisons was earlier genotyped using the Illumina chip according to the manufacturers protocol. Genotypes were called using GenomeStudio (Illumina software) and clustered with other samples not used in this thesis. SNPs with genotyping success rate under 95% were excluded from the analysis.

There are 592426 SNPs with proper success rate on the Illumina 660 chip, of which 555675 were mapping to a unique position in the human genome.

3.10 The pipeline configuration for exome comparison

The analysis is based on two runs of both exome kits. First run was done with 60 bp read lengths and second run with 82 bp read lengths. In addition to running the pipeline for each run separate, the alignments of both runs were combined for both kits and pipeline was run for the combination.

The combination of the alignments was done with the `samtools merge` command. The sorted binary alignment file was merged from run 1 and 2 to create a sorted file for combination. This was then sorted with `samtools sort` and formatted to textual sam format with `samtools view` command. The pipeline was then run without alignment section using the alignment files created as input.

The pipeline was run against the current human genome, the GRCh37 (also known as hg 19). This was downloaded from ensembl (Hubbard et al., 2009). The SNP information file was created from data downloaded from ensembl version 57 human variation version 37a, which has imported the variants from dbSNP (Sherry

et al., 2001) version 131. The exon position file was created from data downloaded from ensembl version 57 using biomart web API (Smedley et al., 2009).

Tool versions used in the pipeline is listed in table 3.4. The x86 version of binary was used if available.

Table 3.4: External software
List of all the external software used in the analysis and their respective versions.

Program	**Version**
bwa (Li and Durbin, 2009)	0.5.7
circos (Krzywinski et al., 2009)	0.52
pindel (Ye et al., 2009)	21st of August 2009
samtools (Li et al., 2009)	0.1.7
velvet (Zerbino and Birney, 2008)	0.7.58

The configuration of the options in pipeline are listed in table 3.5.

Table 3.5: Configuration options
The pipeline configuration options used in the pipeline for exome comparisons. The options which did not affect the results (threading, input file names and sample name) are not listed. The emphde novo max reads option was in first tests 6000000 and then in later executions 0. This is due to memory issues related with the tool which weren't solved by this option.

Option	**Value**
contig size	200
minimum depth	10
coverage depth	20
quality limit hom	0.3
quality limit het	0.8
circos overlap	4
de novo max reads	6000000 / 0

Lastly, an external datafile was used in the comparisons. First was the CCDS exon set for hg19 downloaded from NCBI website June 4th 2010. This was used to calculate mean coverages for genes and exons.

A major part of the exome comparison is based on the target files. There are several target sets for both Nimblegen and Agilent exome kit. See below for name and description of each of the target files.

Given targets

Both the Nimblegen SeqCap Ez and the Agilent SureSelect has their own target area coordinates. For Nimblegen the file with the target area coordinates for hg 19

reference was available after request. This file had 197216 target areas. However for Agilent kit only the hg 18 coordinates were available. These were converted to hg 19 using Galaxy (Blankenberg et al., 2010) lift-over tool. This conversion resulted into loss of 8 of the original 165637 target areas leaving 165629 areas.

Given target areas were combined if there was overlaps between areas to prevent any area from being within each other and sorted accordingly to the requirements in section 3.1 on page 13. This leaves for Agilent 165571 target areas and for Nimblegen 176817 target areas.

Given targets with flanks

In addition to the targets themselves, targets with 100 bp flanking sequences were used as well. The aim here is to see how the coverage of the target area tails. This sized flanking sequences were also used in a presentation by Agilent of their exome kit.

CCDS targets

The consensus coding sequence exome set was selected as a reference exome. This is because both Nimblegen and Agilent aim to target the CCDS exome. The CCDS used is the UCSC (University of California Santa Cruz) CCDS track for hg 18. Hg 18 was selected as it is what the Agilent and Nimblegen used to create the sets and thus it should have more in common with the both Nimblegen and Agilent targets.

After downloading the track with galaxy, it was converted to hg 19 using the lift-over tool and merged in a similar fashion as the target sequences above. The original hg 18 track included 196317 areas. After the above conversion the resulting file included 163627 areas.

Common targets

While both Nimblegen and Agilent aim to target the CCDS exome, neither do this completely. In addition, both had targets which were not in CCDS. Because of this an additional reference set was taken into use in addition to the CCDS reference set. This is called a common targets set.

Common targets were created by overlapping the CCDS with both Nimblegen and Agilent target areas. If both Nimblegen and Agilent target area covers a part of CCDS target, then the area of the CCDS target was included. Note that the actual manufacturers areas do not need to intersect in this situation. For those targets areas within both Nimblegen and Agilent sets which did not match a CCDS area, any intersection was taken.

Chapter 4

Results

The results are divided into two parts. First there is examples of various result files produced by the VCPipeline. These are divided into two sections: main results and supplementary results. Second part is the result of the exome comparisons.

4.1 VCPipeline results

The results of the pipeline come from small variants, coverage, alignment, larger variants, *de novo* assembly and paired-end anomalies sections. Of these, the alignment, small variants and coverage sections provide the main results while larger variants, *de novo* assembly and paired-end anomalies sections provide the supplementary results.

4.2 Main results

4.2.1 Small variants

Small variants produce the main results with regards to the variants. There are tables showing the variant information and three graphs showing the overall qualities.

There are four separate tables for variants.

All the single nucleotide polymorphisms identified from the alignment are listed in the file **variant_table.csv**. This list is filtered with quality checks and also with user given requirements. They are compared to list of known SNPs in dbSNP and if the position matches known SNP, then user is informed of it. Those variants which are in target are flagged and also SNP is flagged if the change is same as in known SNP. Example of this file is in table 4.1. The difference between *Call base* and *VCP call base* is that *VCP call base* is done using quality ratio by conversion script, while *Call base* is original call done by the `samtools pileup`.

Table 4.1: The variant table.
Sequence - Reference sequence entry name. *Position* - Nucleotide position in reference entry. *Reference base* - Base of the reference sequence in the position. *Call base* - Base called by the `samtools pileup`. This can be any of IUPAC unambigious codes. *VCP call base* - Base called by the pipeline. *SNP* - The code of the known SNP or CNV which matches to position. *Allele* - If known SNP or CNV, alleles of the SNP. *Consequence* - If known SNP, the consequence of the SNP. *Gene Transcript Protein* - Possible genes or exons of transcripts or proteins which are in the position of the variant. *Depth* - Depth of the position. *Call depth* - Depth used to make the call. *Reference calls* - Number of reference bases in the position. *Variant calls* - Number of variant bases in the position. *A*, *T*, *C*, *G*, *N* - Number of specific variant nucleotides. *Reference quality* - Sum of quality values of reference bases. *Variant quality* - Sum of quality values of the variant base which has best total quality. *Call ratio* - Ratio between reference calls and variant calls with 0 closing to variant and 1 to reference. *Quality ratio* - Ratio between largest variant base and reference with 0 closing to variant and 1 to reference. *Direction ratio* - Ratio of forward and reverse reads at the position with 0 closing to reverse and 1 to forward. *SNP match* - Flag showing the quality of call match to known SNP, 0 for ok match, 1 for mismatch and 2 for uncertain calls. *Target area match* - Flag to show if SNP hit the possible given target area, 0 for miss and 1 for match. Last two columns *Bases* and *Base qualities*, which show the all the base calls and their respective quality values as characters, were removed from the example as they can be extremely wide.
This example is from the Agilent exome comparison data.

Sequence	Position	Reference base	Call base	VCP call base	SNP	Allele	Consequence	Gene
1	808928	C	T	T	rs11240780	C/T	WITHIN_MATURE_miRNA	ENSG00000234711 (RP11-206L10.12);...
1	1158631	A	G	G	rs6603781	A/G	SYNONYMOUS_CODING	ENSG00000078808 (SDF4)
1	1249187	G	A	A	rs12142199	G/A	SYNONYMOUS_CODING	ENSG00000127054 (CPSF3L)
1	1887019	A	R	R	rs28548017	A/G	STOP_LOST	ENSG00000142609 (C1orf222)

Transcript	Protein	Depth	Call depth	Reference calls	Variant calls	A	T	C	G	N	Reference quality
ENST00000415481;...		20	20	2	18	0	18	0	0	0	65
ENST00000360001;...	ENSP00000353094;...	18	18	0	18	0	0	0	18	0	0
ENST00000485710;...	ENSP00000413493;...	23	23	0	23	23	0	0	0	0	0
ENST00000450874;...	ENSP00000270720	43	43	22	21	0	0	0	21	0	656

Variant quality	Call ratio	Quality ratio	Direction ratio	SNP match	Target area match
560	0.1	0.104	0.65	0	0
543	0	0	0.944444444444444	0	1
706	0	0	0.565217391304348	0	1
661	0.511627906976744	0.498101746393318	0.418604651162791	0	0

The **genotype_table.csv** file lists all the known SNP locations in given target area if there was any sequence coverage at that point regardless of the quality. The format of the table is same as in variant table.

The **indel_variant_table.csv** file shows small indels which are found during alignment. Similar to variants there are quality checks included. These are also compared to the list of known SNPs and user is informed if there is SNP in the position of an indel as not all SNPs listed in the dbSNP are in fact single nucleotide changes but also larger ones including insertions and deletions. Format of the table is similar to variant table.

As with variants the **indel_genotype_table.csv** file lists all the indels in the target area if such were aligned regardless of the quality values. Format is same as in variant indels.

In addition, there is graph showing quality of the SNPs in variant table in **variant_plot.pdf**. It shows call depth, call ratio and quality ratio. Homologous call (where all the variant calls differs from reference uniformly) should have call and quality ratios going towards 0 (reference). While for heterogeneous call (where variant calls are not same but usually divided into two) it should be around 0.5. For example between 0.35 and 0.65. An example of a this file is in figure 4.1.

Last result from this section is the **quality_plot.pdf**. This file has two graphs showing a comparison of quality scores. These can be used to see how well the heterogenous and homogenous variants can be distinguished. One graph is done from variant and other from genotype table. In perfect data set there should be three separate clusters shown in genotype plot. Normally however heterogeneous variants and reference calls don't separate nicely while some separation can be seen between heterogenous and homogenous variants. An example is shown in figure 4.2.

4.2.2 Coverage

Coverage has overall statistics output as well as two coverage files in bedgraph and in gff file format.

The overall statistics of the alignment is in the **alignment_stats_overall.txt**. It gives out several values of how reads are aligned to the target area or into the whole reference. The result is used to get a simple overview of the quality of the sequencing. Results show one fold coverage and a user given fold coverage counts (by default 10). An example of this is shown in figure 4.3.

Second part of the section has two coverage files.

The file that has all aligned reads in gff format is **read_coverage.gff**. Each reference item is divided into user given window size (by default 50) and each record

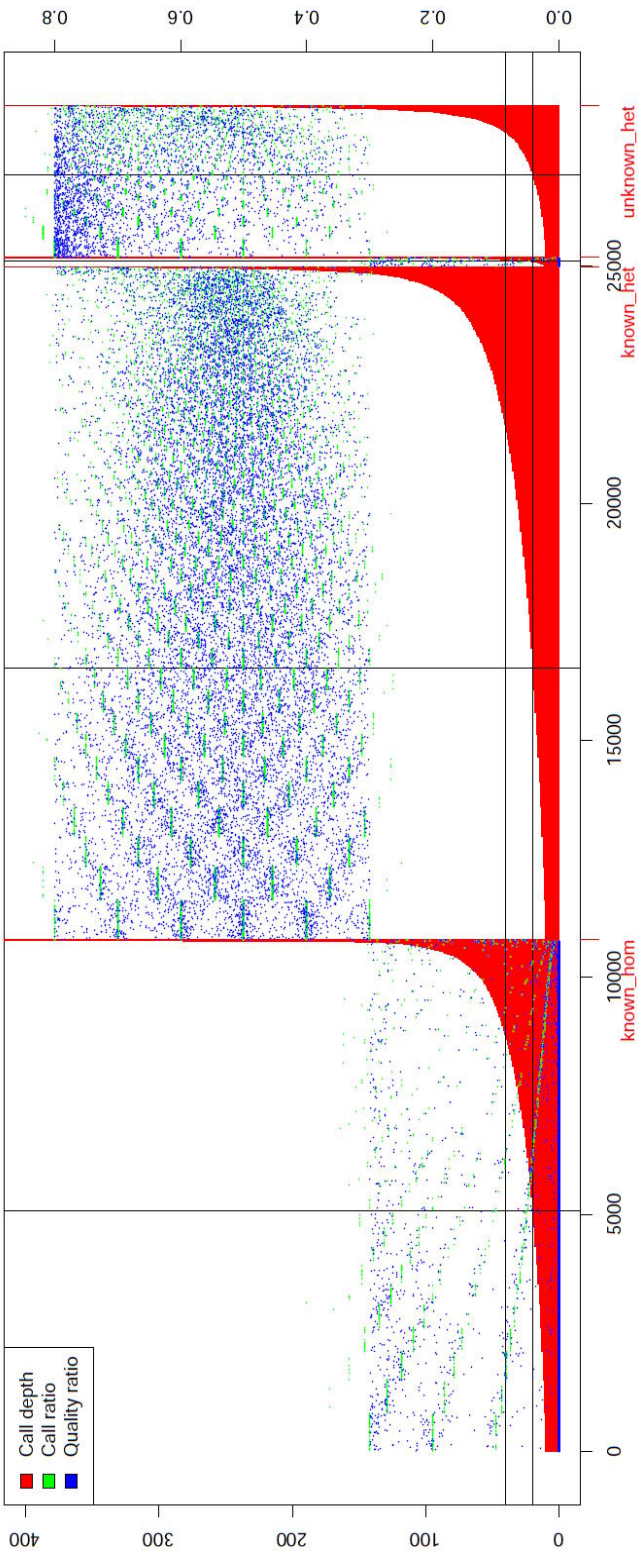

Figure 4.1: Example of a variant plot. The variants are listed on the x-axis, ordered by call depth and grouped in four different categories. First is known homozygous calls, second known heterozygous, third unknown homozygous and lastly is unknown heterozygous calls. On the y-axis are call depth, call ratio and quality ratio. Depth is the number of the reads used to call that variant and its axis is shown in left side of the graph. Both the call and quality ratios vary from 0 to 1 where 0 means that ratio is closer to variant and 1 closer to reference. Their axis is shown in right side of the graph.
The lower horizontal line show the depth which is used as minimum coverage for variant table. Upper horizontal line is double of that. Vertical lines are on the points in which the lower horizontal line intersects with the call depth.
The minimum read depth in this plot is 10. Heterozygous ratio is 0.8 and homozygous ratio is 0.3. Minimum variant table depth is 10. This example is from the Agilent exome comparison data.

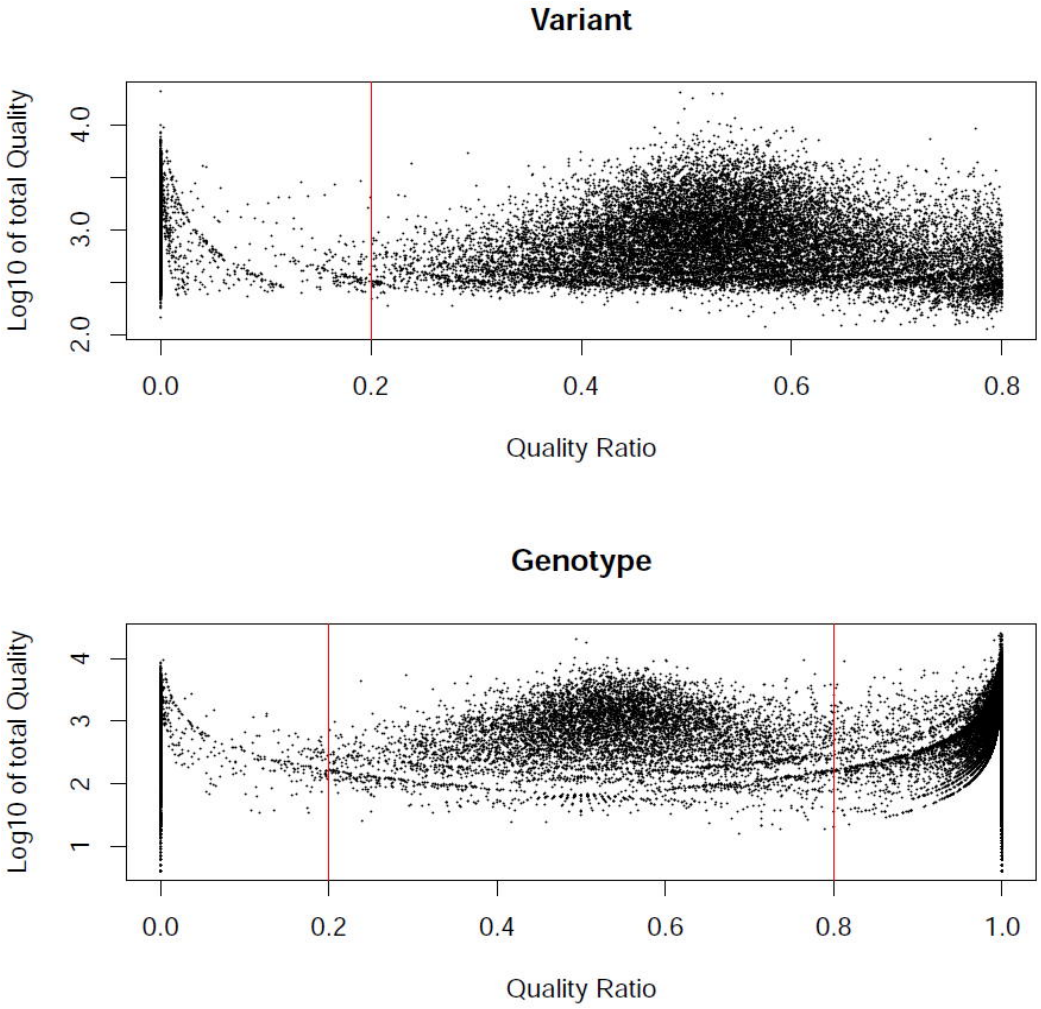

Figure 4.2: Example of a quality plots file. There is two separate plots in the file: variant and genotype. The variant plot is based on the variant table data while the genotype plot is based on genotype table data.
The vertical red lines show the heterozygous and homozygous call limits. The calls in area in the left are homozygous variants. The area in left in variant plot and between the lines in genotype plot are heterozygous calls. The area in right on genotype call is reference calls, this area isn't shown in the variant plot as they are not considered to be variants.
This example is from the Nimblegen exome comparison data.

in the gff file is one window. Windows are ordered by their starting point. The gff score column is used for the coverage within that window. See table 4.2 for example of the start of the read coverage file.

Table 4.2: The read coverage
The coverage is shown in 50 sized windows of genome with windows of coverage 0 skipped. The file is in gff (GFF specification, 2010) file format. The first column is the reference item (in this case the chromosome number). Second column shows the name of the alignment software. Third column has the name of the feature. Fourth column has the window start position and fifth the end position. Sixth column has the mean coverage of the window. Seventh and eight columns are not used and require a dot by specification.
This example is from the Agilent exome comparison data.

1	bwa	Agilent read coverage	10001	10050	1.44	.	.
1	bwa	Agilent read coverage	10051	10100	0.92	.	.
1	bwa	Agilent read coverage	12101	12150	1.46	.	.
1	bwa	Agilent read coverage	12151	12200	3.9	.	.
1	bwa	Agilent read coverage	12201	12250	4.18	.	.
1	bwa	Agilent read coverage	12251	12300	1.08	.	.

Second coverage file **coverage.bedgraph** is in bedgraph format. It has the coverage of the entire genome for each nucleotide. An example of the start of the file is in table 4.3.

Table 4.3: Bedgraph coverage
An example of the start of the bedgraph (BedGraph Track Format, 2010) coverage file. It shows the coverage of the genome with variable sized sections. First row has a track line which describes the data behind it. The first column is the reference item (in this case the chromosome number). The second column is the start point of a section in referencem, the start position is zero-based so 0 means first nucleotide. The third column is the one position after the end point of a section. The last column has the coverage of the section. This example is from the Agilent exome comparison data.

track type=bedGraph name=coverage			
1	0	12104	0
1	12104	12109	1
1	12109	12118	2
1	12118	12119	3
1	12119	12152	4
1	12152	12161	5
1	12161	12168	4

The **target_coverage.pdf** file has a graph which shows the percent of the coverages in the target areas. It can be used to quickly see how good coverage within the target areas can be expected. An example of this graph is shown in figure 4.4.

```
Reads mapping to reference: 91.00% (30398419 / 33403496)
Reference coverage: 12.02% (372032695 / 3095693983)
Reference 10-fold coverage: 1.21% (37398727 / 3095693983)
Target coverage: 91.57% (34455223 / 37627190)
Target 10-fold coverage: 63.61% (23935333 / 37627190)
Reads mapping to target (of mapped to reference): 60.93% (18521684 / 30398419),
(of all): 55.45%
```

Figure 4.3: An example of the alignment counts. The numbers in brackets are as follows. In first line, reads mapped to the reference divided by all reads in alignment. In second line, number of nucleotides covered by at least one read divided by all nucleotides in the reference. In third line, number of nucleotides covered by at least 10 reads divided by all nucleotides in the reference. In fourth line, number of nucleotides covered by at least one read in the target areas divided by all nucleotides in the target areas. In the fifth line, number of nucleotides covered by at least 10 reads within the target areas divided by all nucleotides in the target areas. In the sixth line, number of reads at least partially in the target area divided by all reads mapped to the reference. This example is from the Agilent exome comparison data.

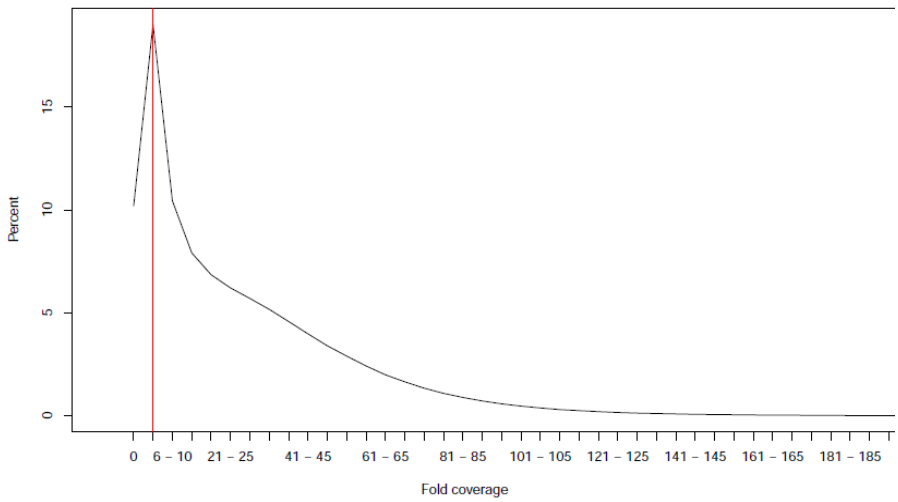

Figure 4.4: The example plot shows a part of the target coverage. The y-axis is percent of the nucleotides and x-axis has the fold coverage groups. The coverage groups are counted per 5 nucleotide, so 0, 1-5, 6-10 etc. The vertical red line shows the largest group. Only up to 400 depth is shown in the actual graph. This example is from the Agilent exome comparison data.

The last result from the coverage section is **heatmap.csv** file. It has a simple table showing the amount of reads at least partially within it for each target area. This can be used to compare several samples if there is noticiable differences in the read amounts per target and if some targets are completely missing from some of the samples. An example of this graph is shown in table 4.4.

Table 4.4: Heatmap table
An example of the start of the heatmap table. First column is the reference item name (in this case chromosome number). Second and third columns are start and end of the target area respectively. Fourth column has the number of reads at least partially in the target area. This example is from the Agilent exome comparison data.

1	30276	30395	430
1	69070	70029	3078
1	367648	368607	2408
1	621085	622044	2233
1	861298	861417	6
1	865533	865745	1
1	866384	866503	10
1	871094	871333	8
1	874405	874524	5
1	874628	874867	4

4.2.3 Alignment

There are two results from this section. One is the alignment file itself and then the other is result of `filter_B.pl` script.

The alignment file is in sam file format. It has all the reads and their aligned positions in the genome. The alignment file is very large, usually over 10Gb, so it's rarely viewed directly and thus no example of it is provided.

The filtering result shows how much from the initial input is filtered out due to quality scores. If in the end of the read there is a block of B quality scores, then the read is trimmed to remove them. If the read length drops below user given limit then the read pair is removed. An example is shown in figure 4.5.

```
Pairs in: 32455200
Pairs out: 31159095 (96.01%)
Trimmed reads: 9512789 (15.26%)
```

Figure 4.5: Example of `filter_B.pl` result. The script removes section of sequence from the end of the read if the quality score for those nucleotides is B. The first line has the count of read pairs in input. Second line has the read pairs for output with percentage in brackets. Third line has the number of reads which were trimmed from end with percentage in brackets (from the output reads).

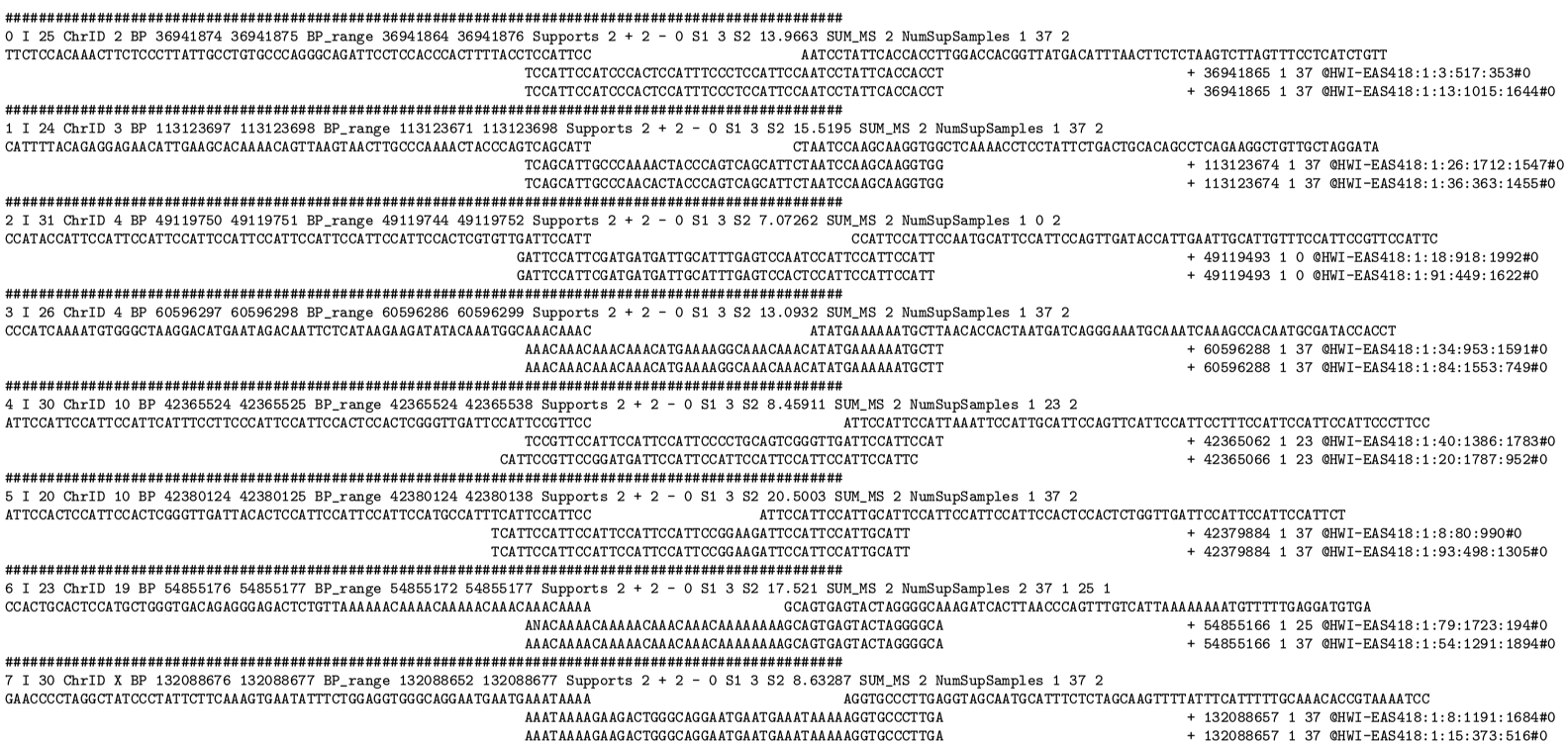

Figure 4.6: An example showing one of the larger variant result files from pindel (Ye et al., 2009) program.
This specific file shows insertions. First line of each entry contains information of the insertion. First number is insertion number. Then there is *I* with number which is the length of the insertion. Next there is *ChrID* followed by reference item value. Then there is *BP* with two numbes showing the position of the insertion. *BR_range* is the same but with breakpoint shift in case of repetitive sequence in reference. Next is *Supports* number + number and - number. These show the support to this with number of reads upstream (+) or downstream (-) of the insertion.
Data is from an association study (Nakki et al., 2010).

4.3 Supplementary results

4.3.1 Large Variants

The large variants produce three files which show various larger variations and their supporting information. The three files created have following types of variations: insertions, deletions and special deletions with non-template insertion around break points. The format is pindel native format. An example of one of the result files is in figure 4.6.

4.3.2 *De novo* assembly

The *de novo* assembly creates a file with assembled contigs. The minimum size of the contigs is given by the user. The file corresponds to the fasta format where on header line there is the node number, length of the contig sequence and coverage of the assembled contig. See figure 4.7 for an example. The result is created in its own subdirectory and in addition there is supporting files which can be looked into for more information.

```
>NODE_12_length_180_cov_4.200000
GAACCGCTCTTCCGATCTTTTAATCAGATGATGGTGGTGCTACCCCAGTACAGGATGAAC
GGGATTCAGGGTCAGACGGTGAGGATGATGTAAATGAGCAACACTCCGGATCAGACACTG
GAAGTGTAGAACGTCATTCAGAGGTATTGGCTAACCAACATCTTTGGGAGGGCTTTTCCC
TTGCGTTCTGTGAGATCTGTTATA
>NODE_53_length_228_cov_47.285088
GTTGTTTTCTAGCAGTGACAAGTTCACTTTGAATCAGGTTTGAACTTGACAATTTACTGT
CTTCCTCATTGAATTCCTCCTTGCACATTTCTGCTTTATCTCATCTACACAGAAGTGATC
CAATATTTAGCTATAGAGCTATATTAGTTAAGAAGGTATTTTTAAAGTAAAATTTGTAGG
TTTTTAGCTTAGTCTCCATTTAAAATATGTTCTGTTTTCTTAACTTCAGGATATGTGTGT
AGTTTGTGGCAG
```

Figure 4.7: A start of an example result of the *de novo* contig assembly. The assembled contigs are in fasta format. The contigs are assembled from read sequences which were not aligned to the genome. This example is from the Agilent exome comparison data.

4.3.3 Paired-end anomalies

The paired-end anomalies section produces pictures and gff tracks.

In picture there is a circle with green sections showing different reference sequence entries, which in case human genome would most likely be chromosomes, with red lines starting from the entry for which the picture is drawn. Each red line represents multiple pairs of reads, which overlap each other, pairing anomalously within the genome. Either the pairs' ends are in separate reference items or too far apart within

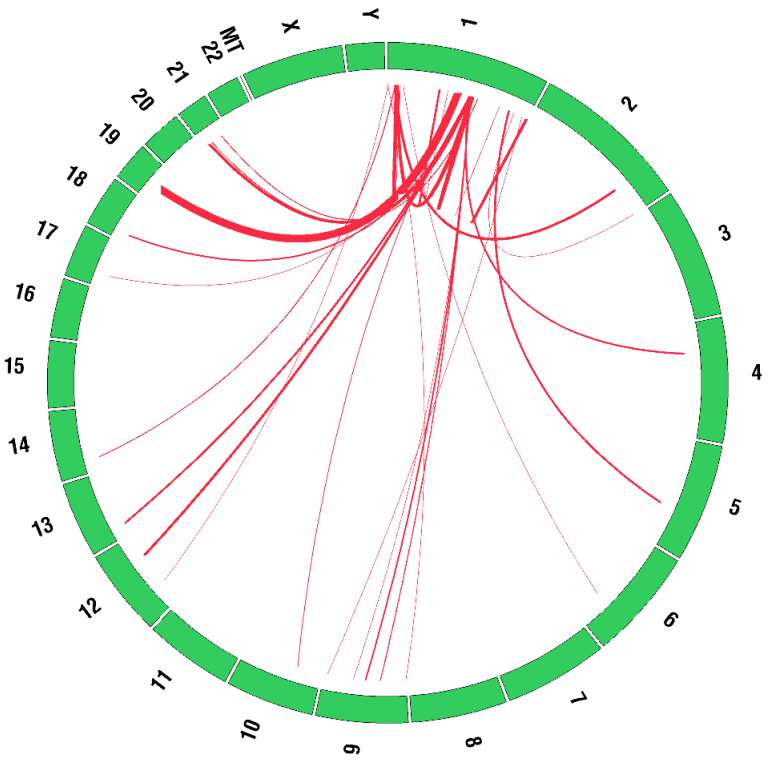

Figure 4.8: Example of the anomalies drawn by Circos (Krzywinski et al., 2009) from paired-end alignment. This is for the reference sequence entry named '1' (meaning chromosome 1), which can be seen from the picture as all the lines have one end in entry 1. Each line represents a multiple overlapping pairs of reads which start from one chromosome and end in other or are too far apart of each other in same chromosome. The thickness of the red line is relative to the amount of overlapping read pairs. This example is from the Agilent exome comparison data.

the same reference item. The thickness of the line is relative to the amount of pairs in the anomaly. An example of the Circos plot is shown in figure 4.8. Anomalies shown in picture can be artefacts of the sample preparation, sequencing or possibly true translocations or larger deletions. The anomalies result can also be used to detect possible sequence input mistakes and mixups.

The gff files then show more detailed information. For each of the reads exact position, how many other reads overlap that read and the position of the pair is described. See example in table 4.5.

Table 4.5: The anomalous pairs
The first column is the reference item (in this case the chromosome number). Second column shows the name of the alignment software. Third column has the name of the feature. Fourth column has the window start position and fifth the end position of this end of the read pair. Sixth column has the amount of overlaps that this read has. Seventh and eight columns are not used and require a dot by specification. Ninth column has the start and end position of the other end of the pair as well as the read pair name (which is removed from below). The rows are in pairs so the other end of the pair is in next row.
This example is from the Agilent exome comparison data.

1	bwa	set1-agilent odd paired	8716175	8716233	3	.	.	22567021, 22567079:
15	bwa	set1-agilent odd paired	22567021	22567079	3	.	.	8716175, 8716233:
1	bwa	set1-agilent odd paired	8716178	8716236	3	.	.	22567022, 22567080:
15	bwa	set1-agilent odd paired	22567022	22567080	3	.	.	8716178, 8716236:
1	bwa	set1-agilent odd paired	8716182	8716240	3	.	.	22567014, 22567072:
15	bwa	set1-agilent odd paired	22567014	22567072	3	.	.	8716182, 8716240:

4.4 Exome comparison results

This section presents the results from the exome comparison.

4.4.1 Reads

The amounts of reads in each run at various stages are listed in table 4.6. The amount of nucleotides are not listed for each step as filtering done in alignment (`filter_b.pl` script in section 3.3 page 18) can trim the read.

Table 4.6: Read counts
The table lists how many reads were produced and aligned. *Produced* - the number of reads produced by the Illumina software. *For alignment* - the number of reads that are input to the alignment after all the filtering is done. *Aligned* - the number of reads which are aligned into the reference sequence. *On target* - the number of reads of which at least one nucleotide is within the target area.

Manufacturer	**Run**	**Produced**	**For alignment**	**Aligned**	**On target**	
					CCDS	Own
Agilent	Run 1	35426712	31027419	29945945	15883486	18147078
Nimblegen	Run 1	37662148	30742570	30665305	20957572	24844142
Agilent	Run 2	59817166	52304467	46956740	26715060	29697854
Nimblegen	Run 2	64910400	53080289	52671092	39343229	44455439
Agilent	Comb	95243878	83331886	76902685	42598546	47844932
Nimblegen	Comb	102572548	83822859	83336397	60300801	69299581

4.4.2 Coverage

Coverage section deals with coverage across the targets and genes. Only some results of the pipeline are shown. Others are derivates showing the comparisons.

The coverage values from both sequencing runs show Nimblegen preforming more efficiently, meaning larger percentage of reads align against reference, with larger amount of target having 20 fold coverage. This is shown in all targets. As an example the figure 4.9 shows a counts file from both Agilent and Nimblegen using CCDS target. A graph showing the 20 fold coverages of all target areas for both kits is shown in figure 4.10.

a

```
Reads mapping to reference: 89.78% (46956740 / 52304467)
Reference coverage: 19.42% (601219016 / 3095693981)
Reference 20-fold coverage: 1.21% (37427019 / 3095693981)
Target coverage: 92.61% (25763572 / 27819612)
Target 20-fold coverage: 60.71% (16889307 / 27819612)
Reads mapping to target (of mapped to reference): 56.89% (26715060 / 46956740),
(of all): 51.08%
```

b

```
Reads mapping to reference: 99.23% (52671092 / 53080289)
Reference coverage: 9.94% (307803672 / 3095693981)
Reference 20-fold coverage: 1.23% (38102971 / 3095693981)
Target coverage: 89.93% (25017284 / 27819612)
Target 20-fold coverage: 66.58% (18523002 / 27819612)
Reads mapping to target (of mapped to reference): 74.70% (39343229 / 52671092),
(of all): 74.12%
```

Figure 4.9: Exome comparison counts result. The two examples are from the results of the sequencing run 2 with CCDS target area. Figure (**a**) is from Agilent and (**b**) from Nimblegen.

To see how the targets are covered in more detail a graph can be done from target coverages. This will show how balanced the target coverages are in both. The CCDS and the kits' own targets were used for this. The resulting graph is shown in figure 4.11.

In order to get coverage of known genes, the datafile with CCDS exons was used. That was then combined with the bedgraph coverage files from each sequencing run to produce mean coverage of all genes in each of the runs. The resulting genes were sorted according to the coverage and plotted to show the overall view on how well the genes are covered. The plot can be seen in figure 4.12.

The files created above were then used to gather the amount of transcripts completely covered with various fold coverage. For transcript to count completely cov-

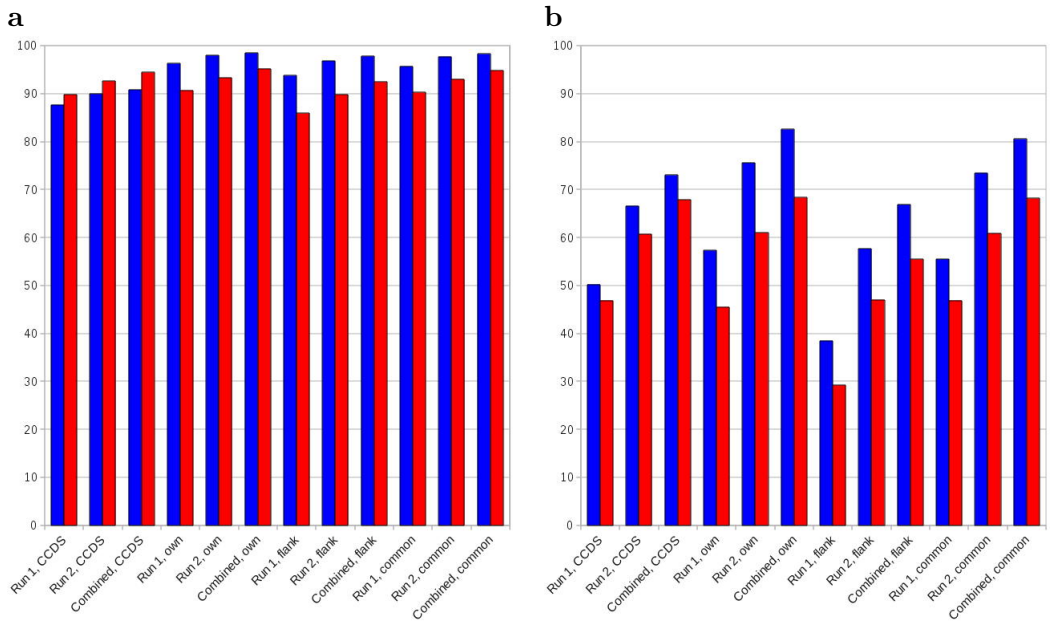

Figure 4.10: Counts comparison graph. The bars show the percentage of target areas covered within the sequencing runs. Target areas are as follows: *own* - kits' own target, *flank* - kits' own target with 100 bp extra on both flanks, *common* - areas which is shared between the kits and *CCDS* - ccds (Pruitt et al., 2009) exons. The blue coloured bars are for Nimblegen and red coloured bars for Agilent. Figure (**a**) has one fold coverage while figure (**b**) has 20 fold coverage.

ered, all exons within the transcript had to have the required coverage. This was done for combined coverage of both runs for both kits. In total there was 23746 transcripts in the CCDS file. The list of transcript coverages can be seen in table 4.7

Similarly we can go through the exons and see how many of them are covered with various fold coverages. There is total of 185508 unique exons in the CCDS file (after removing the exons which were in different transcript but in same position as another exon), of which Nimblegen covers 133546 exons and Agilent 120923 exons. Of these, 108466 exons are shared. This means that Nimblegen has 25080 unique exons while Agilent has 12457 unique exons. The list of exon coverages can be seen in table 4.8.

Heatmaps can also be used to see how similar the kits are. Using the common and ccds target we can compare the areas which both Nimblegen and Agilent should cover. Since the read amounts fluctuate between sequencing runs, the comparison of read counts does work directly. By counting areas with no reads we should see equal number of deletions as long as the target should be covered. With common target, there is 2634 target areas with no reads in Agilent and 606 in nimblegen. Of these 528 are same. This leaves 2106 unique areas with no reads in Agilent and

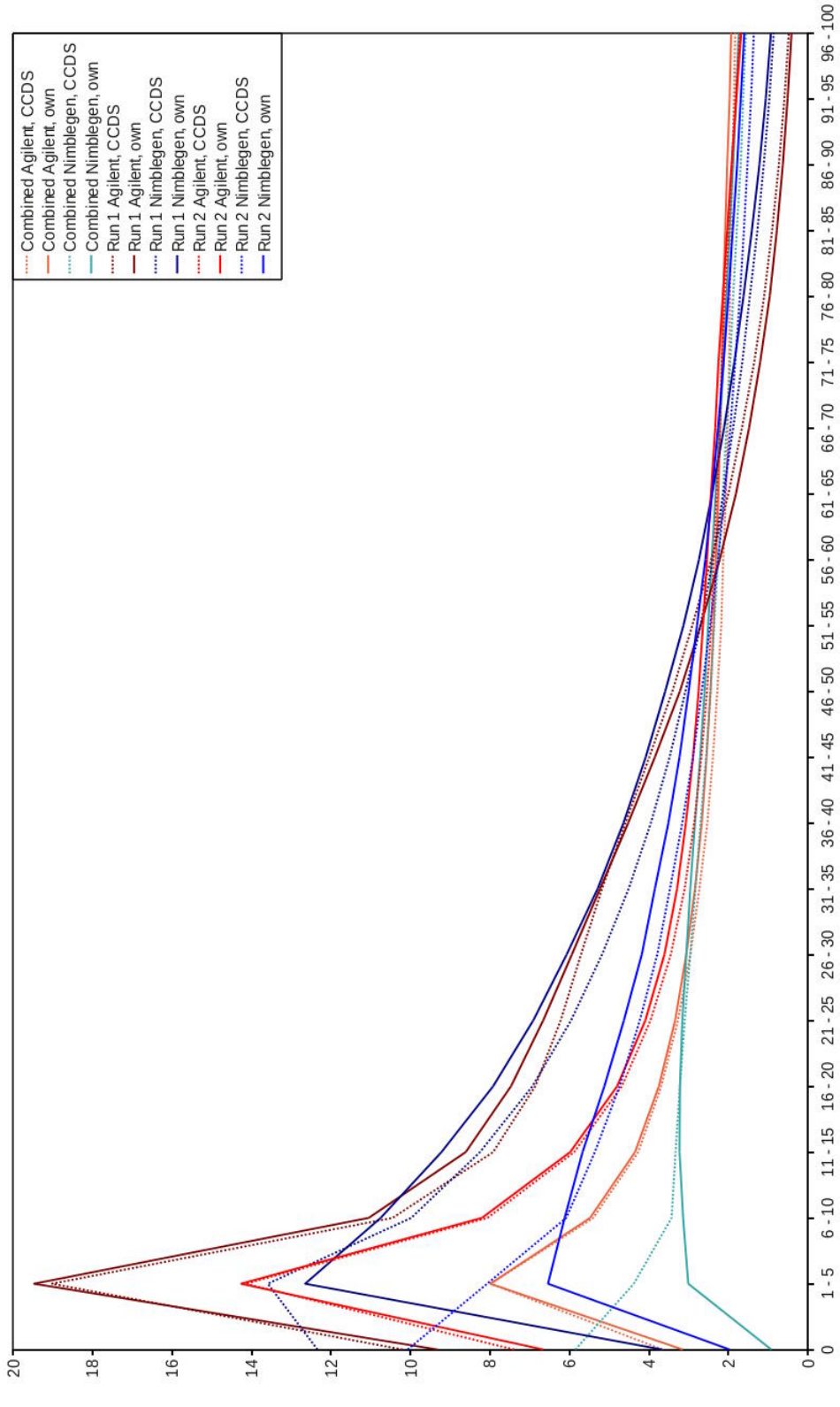

Figure 4.11: Target coverage. The graph shows how much of the target is covered by what read depth. In x axis there is coverage and on y axis percentage. Coverage is in 5 sized bins.

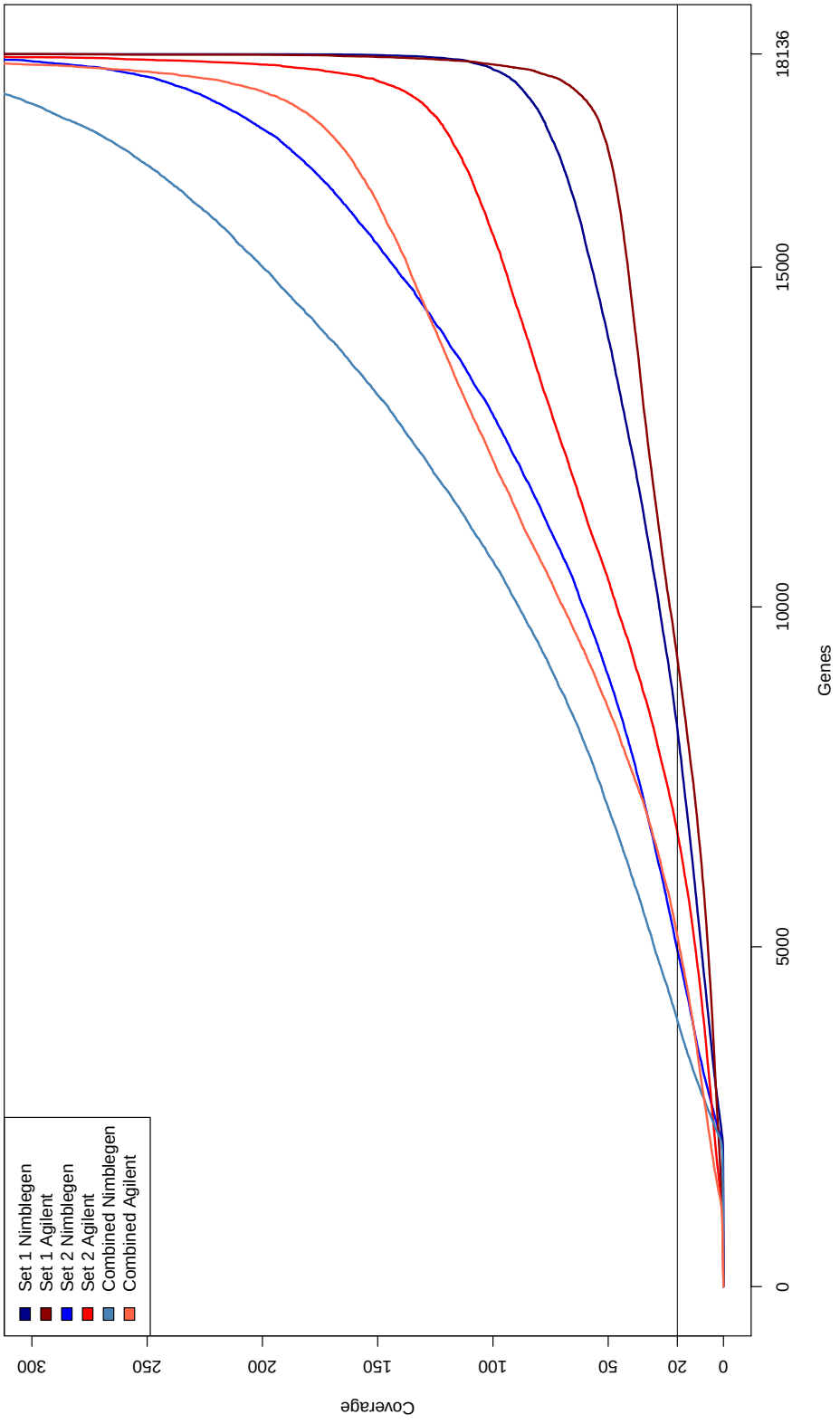

Figure 4.12: Genes coverage. The graph shows the mean coverage of the genes for each sequencing run and the combined results of the two runs. The maximum coverage of 300 is shown with total of 18136 genes. The Agilent has wider coverage of the genes with small coverages but the Nimblegen has more genes with good coverage.

Table 4.7: Transcript counts
The table lists the count of transctripts completely covered by a required coverage. The transcripts were taken from CCDS (Pruitt et al., 2009). The coverages were counted from the combined data.

Manufacturer	**Coverage**	**Transcript count**	**Transcript %**
Agilent	20	6621	27.88
Nimblegen	20	9652	40.65
Agilent	10	8899	37.48
Nimblegen	10	12686	53.42
Agilent	5	11273	47.47
Nimblegen	5	15168	63.88
Agilent	1	16201	68.22
Nimblegen	1	18134	76.37

Table 4.8: Exon counts
The table lists the count of exons covered by a required coverage. The transcripts were taken from CCDS (Pruitt et al., 2009). The coverages are counted from the combined data.

Manufacturer	**Coverage**	**Exon count**	**Exon %**
Agilent	20	120923	65.18
Nimblegen	20	133546	71.99
Agilent	10	136627	73.65
Nimblegen	10	143062	77.12
Agilent	5	148064	79.81
Nimblegen	5	148083	79.82
Agilent	1	163249	88.00
Nimblegen	1	153927	82.98

78 in Nimblegen. With CCDS target, there is 3481 target areas with no reads in Agilent and 11219 in Nimblegen. Of these 1056 are same. This leaves 2425 unique target areas with no reads in Agilent and 10163 in Nimblegen.

4.4.3 Variants

With variants most comparisons are done with the combined dataset. The comparison are done between the variant table listing the found variants or genotype table listing all the known variant positions. In addition, the quality is evaluated with external SNP-chip using the genotype calls created by using the Illumina Human 660 Quad chip.

First the found variants are compared.

First we compare the methods against a suitable reference to see how much difference there is in variant calling between them. For this the CCDS and common targets are suitable. CCDS because the target which is the aim for both kits and common because that is the area where their targets actually overlap. Within the CCDS targets, there were 12547 variants identified by the Agilent capture and 13406 by the Nimblegen capture. A total of 10526 variants were in the same position. This means that Agilent identified 2021 (16.11%) and Nimblegen 2880 (21.48%) unique variants within the CCDS target area. For the common target area, the Agilent identified 11326 and the Nimblegen 13444 variants. 10571 of those were in the same position. So Agilent identified 755 (6.67%) and Nimblegen 2873 (21.37%) unique variants within the common target area.

Next the SNP calls were compared with genotypes produced by using the Illumina SNP-chip. Of the SNPs represented on the chip, 7381 positions were covered by the sequences obtained from the Nimblegen capture and 7464 from the Agilent capture with a minimum of 10 fold coverage. From these positions, the chip was unable to make the call in 184 sequenced by Nimblegen and 224 sequenced by Agilent. 7438 (99.65%) of the VCP calls from the Nimblegen capture agreed with the genotype call from 660 chip, while the numbers were 7321 (99.18%) for the Agilent capture. Further, 5899 SNP were covered by all three methods. Of these VCP calls from Nimblegen and Agilent captures agreed on 5862 (99.37%) SNPs, and 5710 (97.41%) of them also agreed with genotyped call. Of the 37 discrepant calls between the two capture methods, 21 of the Nimblegen capture derived calls agreed with genotype call while the corresponding number for Agilent capture was 12. Chip was unable to call 4 of those. Differences between the two kits were whether the call heterozygous or homozygous.

Then the known variant positions are compared.

There are 156811 known variants in the CCDS target. Of these there are 153284 within the Agilents own target area and 134391 within the Nimblegens own target area. With a minimum of 10 fold coverage Agilent has 115188 SNPs (75.10%) and Nimblegen has 123498 (91.19%) SNPs sequenced.

From the own targets of the Agilent and the Niblegen, there were 158604 known variant positions within both capture methods. Within these positions they agree on 149757 (94.42%) positions. Of these Agilent has 126698 (79.89%) positions and Nimblegen 146671 (92.47%) positions with at least 10 fold coverage. With the minimum coverage of 10 there is 117188 same calls.

Then the SNP calls were compared to Illumina SNP-chip. From chip, 9691 SNP positions were covered by the sequences obtained from the Agilent capture and 9803 SNP positions from the Nimblegen capture with minimum of 10 fold coverage. Chip was unable to call 384 positions sequenced by the Agilent and 369 sequenced by Nimblegen. 9115 (97.94%) of the VCP calls from the Agilent capture agreed with genotype call from the 660 chip, while the Nimblegen agreed with 9194 (97.46%) calls. Further, 7826 positions were covered with all three methods. Of these VCP calls from Nimblegen and Agilent captures agreed on 7509 (95.95%) SNPs, and 7215 (92.19%) of them agree with genotyped call. Of the 317 discrepant calls between the two capture methods, 128 of the Nimblegen capture derived calls agreed with genotype call while the corresponding number for Agilent capture was 166. Chip was unable to call 20 of those and in 3 cases all methods gave different result.

Then as last part of this section, some comparison are done between the runs of the same method to try to see how stable the sequencing results are. There was 28386 SNPs found Agilent on the run 1 and 42166 on the run 2 with minimum of 10 fold coverage. Of these, 25655 (90.38% from run 1) were in both. The 2731 SNPs which are missing from run 2, 2068 (75.72%) were not covered with 10 fold coverage. The remaining 663 were not called by default. When manually retrieved and called 376 (13.77%) were same and 287 (10.51%) different. With Nimblegen there was 29225 SNPs found on the run 1 and 41135 on the run 2 with minimum of 10 fold coverage. Of these, 26513 (90.72% from run 1) were in both. The 2712 SNPs which are missing from run 2, 2197 (81.01%) were not covered with 10 fold coverage. The remaining 515 were not called by default. When manually retrieved and called 238 (8.78%) were same and 277 (10.21%) different. Further details of the effect of increasing the amount of reads (and lengths) to the variant calling can be seen in table 4.9.

Table 4.9: The effects of read amounts to variants
The comparison below is done to show the effect of read amouts with variants. Run 1 provided little over half of the reads of Run 2. Combined has both runs so there is little under 50% more reads than in Run 2. All calls have minimum fold coverage of 10. The percentages show the increase to previous run counts for the same method. The found variants are listed in table (**a**) and the known SNP positions are listed in table (**b**).
Run - the run and method. *Total SNPs* - the amount of SNPs listed. *In SNP chip* - count of SNPs which are in SNP chip and sequenced. *Genotyped* - count for SNPS which chip was able to call. *Same* - count which matches the SNP chip genotype calls

a

Run	Total SNPs	In SNP chip	Genotyped	Same
Run 1 Agilent	28386	4862	4730	4621
Run 2 Agilent	42166 (149%)	6598 (136%)	6419 (136%)	6384 (138%)
Combined Agilent	49745 (118%)	7587 (115%)	7364 (115%)	7321 (115%)
Run 1 Nimblegen	29225	5320	5232	5146
Run 2 Nimblegen	41135 (141%)	6931 (130%)	6774 (129%)	6661 (129%)
Combined Nimblegen	47805 (116%)	7665 (111%)	7481 (110%)	7438 (112%)

b

Run	Total SNPs	In SNP chip	Genotyped	Same
Run 1 Agilent	189457	7672	7362	7269
Run 2 Agilent	195662 (103%)	8988 (117%)	8627 (117%)	8532 (117%)
Combined Agilent	200286 (102%)	9691 (108%)	9307 (108%)	9115 (107%)
Run 1 Nimblegen	182651	8370	8101	7988
Run 2 Nimblegen	186860 (102%)	9494 (113%)	9148 (113%)	9044 (113%)
Combined Nimblegen	187828 (101%)	9803 (103%)	9434 (103%)	9194 (102%)

4.4.4 Paired-end anomalies

A selection of paired-end anomalies pictures are collected to figure 4.13. These are a small but representative selection of what can be seen in the images overall. (a) and (b) show most anomalies in both. (e) and (f) is an example of multiple situation where Nimblegen doesn't show any lines while Agilent does. (c) and (d) are then example of something in between the two pairs detailed above.

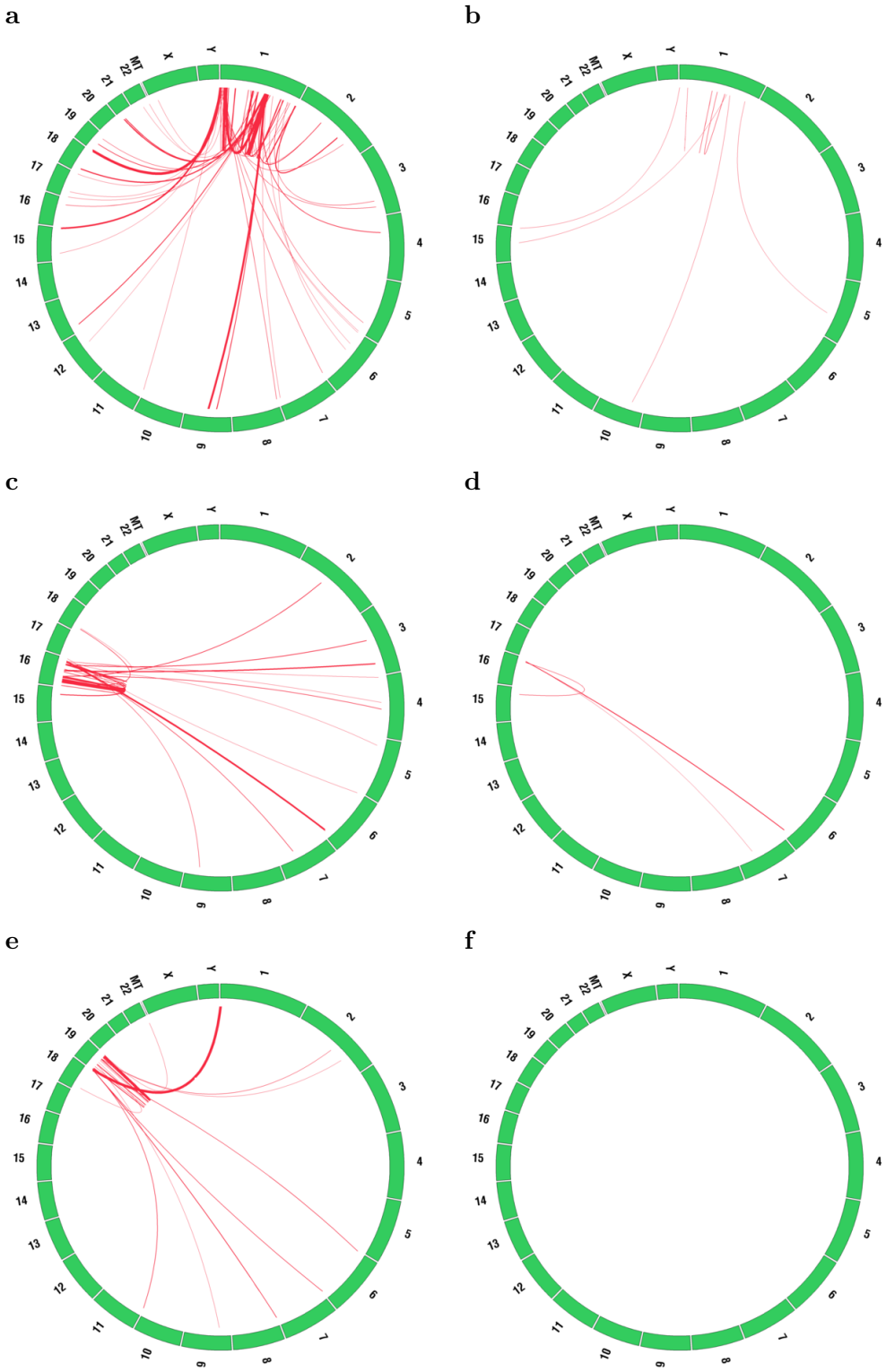

Figure 4.13: Paired-end anomalies comparison. Each picture represents one chromosome. Each red line represents a multiple overlapping pairs of reads which start from one chromosome and end in other or are too far apart of each other in same chromosome. The (**a**), (**c**) and (**e**) are from Agilent and (**b**), (**d**) and (**f**) are from Nimblegen. The images are from combined data set.

Chapter 5

Conclusions

5.1 VCPipeline results

Overall there is a good set of results to start the further analysis from, as could be seen in the exome comparison results in section 4.4 on page 40. The main results include several results from alignment to variations (such as SNPs and indels) and coverage as well as quality control aids. Supplementary results then extend from this to few directions (such as trying to recover information from unaligned reads). Extending the pipeline to give complete results for the user (eg. the person who ordered the sequencing) can be very difficult as the needs can vary greatly depending on the project.

In addition to the results for the user, there are several different quality metrics for the sequencing laboratory. These will help to make sure that there are no false results given out and may help with further development of methods. Some of the results are as useful for the user as for the sequncing laboratory. The conclusions below are considered from the point of the user.

The pipeline described in thesis was designed with an assumption that the sequencing produces a one pair of sequence files as a result of single paired-end sequencing of a single sample. The Illumina GAII sequencer produces a pair of sequence files for each lane, so the assumption is that each lane has one sample. Thus, the pipeline handles the basic situation where every lane is separate sequencing event. However there are two exceptions to this. Multiplexing and merging.

In multiplex situation there is multiple samples in one lane and there is an index sequence which can be used to separate the samples as explained in section 1.2.2 on page 6. At the moment there can be up to 12 separate indexes per lane leading to 96 separate samples in the 8 pairs of sequence files produced. While the separating the samples could be done manually, running the pipeline manually for 96 times is

not very practical. As an improvement, the pipeline could support multiplexing so that it has to be executed only once.

Merging is samilar situation, but looked from another direction. In merging a single sample is sequenced in multiple lanes leading to the need of combining multiple sequencing results into one. Running pipeline manually in this case would require running the alignment section multiple times. Then manually merging alignment results which would then be used as input to the rest of the pipeline. Again to reduce the manual work this should be supported so that pipeline should executed once.

5.1.1 Main results

The first files to look in the results are the alignment stats. This file will tell whether the sequencing and the possible target capture have succeeded. Preferably more than 80% of reads should align to genome in a good result. If the sequencing is a targeted sequencing, then alignment stats can also be used to confirm that the targeting has worked. With poor target enrichment, the higher fold coverages are often very low (for example 20%) for target while proper number of reads have been aligned against the reference (more than 80%).

The alignment files are the most important result for the user even though some users might not look at them directly. They present an interesting problem with regards to visualisation. There is small number of programs which can open the files as they are multiple gigabytes in size (such as the alignment files). One such program is the Integrative Genomic Viewer (Broad Institute, IGV). An alternative would be to setup an account in UCSC Genome Browser (Kent et al., 2002) and have a local server to serve the data on-demand.

The main tables of interest are the variant table and the indel variant table. These provide the main results with regards to the variants (SNPs and small indels) in a tabular format and can thus be used in further analysis. However the pipeline could be improved to allow some form of visualisation of the variants as well. For example if alignment is visualised, then the called variations should also be visualised with it to allow the visual inspection to which of the variations shown in the alignment are also called.

The quality graphs and variant plots are used for quality assessment to identify potential major skews in variant calling. This part of the pipeline results are mainly used by the sequencing laboratory before releasing the data to user. This is important to user in order to prevent badly created data from being used in analysis and causing incorrect conclusions.

Coverages are shown in two different formats. Bedgraph format is more accurate representation of coverage while many locally installed tools might work better with gff formatted files as it is an older format. Coverage results can be looked in specificity if there is a gene or set of genes or some other areas of interest to make sure that those were sequenced properly.

Heatmap table can also be used to make sure that all the target areas were sequenced. With combination of coverage and heatmap files, the user can try to detect deletions or other structural variations which would result to deletion of a target area. The heatmap however only tells how many reads touched the area. It might be worthwhile to change the format a bit by reporting the amount of nucleotides the read actually covers instead. This should show more accurate value of it's coverage.

5.1.2 Supplementary results

Supplementary sections each provide a single type of result.

For larger variantion there is three separate files which are used to identifying different kinds of variations. These are an addition to the indel results provided above and cover larger indels than what the earlier result had. Each of these are usable by themselves.

Paired-end anomalies result should be used in conjuction with coverage results. There should not be a large difference in the support of those and coverage at that position. Currently the score given in the gff entry is half of the support (rounded down to full integer). This is due to reasons of circos drawing and is something that should be improved. Additionally, some filtering to these results should be done to reduce the false positives. For example comparing the positions to coverage and discarding by user given ratio as at least 25 or 50% of the coverage should be in anomalies for them to be valid.

Currently the *de novo* assembly section is mostly turned off in the pipeline (by setting the read maximum to 0). This is due to reasons of memory use by the assembler. An assembler, such as ABySS (Simpson et al., 2009), which can be parallellised should be taken into use. This would help with the memory loads.

5.2 Exome comparison

For final analysis, additional capture experiments and additional sequencing runs would be needed. The data presented here is based on one capture experiments per capture method and two sequencing runs. However further experiments with these capture methods have provided comparable results (results not presented here).

5.2.1 Reads

The read statistics show that in general Nimblegen capture protocol provides more reads than the Agilent one, but filtering seems to remove more reads from the Nimblegen than from the Agilent. Thus, similar amount of reads enters into alignment. However for both runs the Nimblegen capture seem to align somewhat more efficiently on the reference and on target areas as can be seen in table 4.6 and the figure 4.9. After filtering and removing the duplicates both capture methods provide high quality template libraries as approximately 90% or more of the produced sequence align to the reference sequence.

5.2.2 Coverage

Since there can be sequencing errors in individual reads a high coverage is needed for the areas of interest as many errors on same position by multiple reads is highly unlikely. The results show two separate sets of data within coverage comparisons. First is the target area comparison and second is gene related comparisons.

Target coverages can be seen in figure 4.10 and graphs of figure 4.11 with regards to coverage statistics and also in target heatmaps. The coverage statistics show that Nimblegen captures provide more high coverage data while sequences from Agilent capture are less concentrated on target areas. The individual statistics indicate that the Agilent capture covers more of the genome outside of the target than Nimblegen.

Target coverage graphs allow a more detailed look at the target coverages. An optimal coverage would be approximately A perfect target coverage graph should have spike with good coverage, for example in between 20 and 50 coverages, and nothing outside it. Agilent capture methods show a spike in 5 - 10 coverage bin and also correlates with the lower 20 fold coverage in statistics. This implies that while some of the probes are very efficient in capturing the genomic DNA, other probes have perform less efficiently. The heatmap comparison with no reads on common areas seem to support this as there is more areas with no reads in Agilent than in Nimblegen. The graphs of the Nimblegen captures does not seem to have any major spikes and the combined data is more flat any other graph. This is somewhat unexpected as coverage statistics showed that the Nimblegen provide more areas with sufficient coverage. Further Nimblegen seems to have a bit larger percentages in high coverage depths. This can be seen in the run 1 graphs in the target coverage table. This would imply that Nimblegen has too many probes for areas which get very high coverage, over 100 fold coverage in this study.

The fact that the Agilent capture method covers more of the CCDS target area than the Nimblegen can also be seen from the results. The coverage of its own target

seem to correlate with CCDS unlike with Nimblegen captures. Then in the coverage statistics the Agilent with one fold coverage has better CCDS target coverage than the Nimblegen. And finally, in the heatmaps the Nimblegen has much more areas with no reads.

Second interest of looking at the coverage is to see how well genes and their products are covered. The results are in two tables: the transcript coverages in table 4.7 and exon coverages in table 4.8. In addition, there are genes coverage graphs in figure 4.12.

For genes and transcripts, all the results are showing similar information. The Agilent is covering more genes with low sequence coverage while the Nimblegen is covering higher percentage of genes with high coverage. At the exon level the Agilent is covering more exons with read depth of 1 while more exons are covered with depth 10 or more with the Niblegen capture method. Gene coverage graph reflects the exon results. In exons and genes the results show the mean coverage of the area in question, all exons of the genes are checked counting the mean over all to get the coverage. However for transcripts, multiple exons are checked separately to fulfill the criteria and only transcripts which has all exons above the limit are considered to be covered properly. This might explain the difference in transcript result where Nimblegen is better even in low coverages. So Agilent capture seems not to be able to cover all the exons in transcripts even if it covers more individual exons than the Nimblegen capture.

5.2.3 Variants

There is very little variation between the two methods with regards to the comparisons between the CCDS and common targets. Nimblegen has few more variants in those targets but the difference is not significant. Only noteworthy difference is when comparing known genotypes. The Nimblegen has over 10% more of known SNPs than the Agilent. Comparison of SNP posisions which both capture methods cover has almost no difference (more than 99% are called same).

When doing the quality comparisons against the Illumina SNP-chip there is again very little difference between the methods. Both methods seem to produce high quality variants as most calls (more than 95% in all comparisons) which intersect with chip SNPs are the same. It is noteworthy to mention that both of the the sequencing methods presented were able to produce variant calls on positions which the chip was unable to produce a genotype call.

Sequencing runs one and two are not directly comparable in most cases, because the amount of reads produced as well as size of the reads are much lower with both

kits on run 1 compared to run 2. Two comparisons are made however. First, to find out how many of the variants are retained between runs and secondly how many new variants was found due to increased data.

Since run 2 produced much more sequence, it would be assumed that higher number of the variants would be retained than in equal sized runs. With both methods the retain rate of variants between the two runs was about 90%. Most of those which were lost with both methods were due to not having sufficient depth in run 2 (over 75% with both). Larger part of those which did have sufficient coverage were calls which were near the 0.8 quality limit and were called reference. They in most cases did have same variant bases as in run 1. However in run 2 the quality of the variant bases were either too low or there was too few variant calls. Many of those variants were near the 0.8 limit in the run 1 as well. Smaller part of unretained variants would have been clear variants by only looking at the quality ratio. However the algorithm which detects the variants before quality filtering is applied, did not call them for unknown reason.

In addition, the table 4.9 can be used to see how much more variants are called by increasing the data available. With variants called the differences are quite large between runs 1 and 2 as it should be. However combining the two runs results in substantially smaller gains. This is most likely due to the targeting. As both runs aim to sequence the same area, the saturation in many areas does not give more information on it. Only the targets with smaller coverage are aided with this. It also suggests that some of the areas within the targets are very hard to capture adequately and thus adding more and more sequencing lanes will not help to increase the coverage of those areas.

There seems to be practically no difference between the two methods in sequencing accuracy. For variant calling both are equally good. For further validation, a number of discrepant calls should be sequenced using the Sanger sequencing. The main effect to variant calling comes from coverage. If user is interested in variations in some specific genes, then it makes more sense to make sure that the method used covers the gene in question properly.

5.2.4 Paired-end anomalies

The main intention with the paired-end anomalies is to find translocations. However since the data being analysed is exome sequencing data, there is little chance of actually finding them. For exome sequencing to find a translocation, the break point needs to be in an exon and not in an intron or between genes. Since exome that these kits sequence is a small part the genome, it is not very likely to show

true translocations. The picture can however be used to see how much noise the capturing methods produce.

The paired-end anomalies pictures (shown in figure 4.13) show less anomalies for the Nimblegen capture. Only few images were selected into the figure, but there was no single pair where Agilent would have had less spreading of the sequence when compared to Nimblegen. Only chromosome with no lines for both was mitochondria.

Acknowledgements

I would like to thank FIMM for providing me the opportunity to do my thesis. The process has been challenging but at the same time very rewarding. Thanks especially to Janna Saarela and Pekka Ellonen for your guidance and expertise. Pirkko Mattila and Maija Järvinen for helping the proof reading and Anna-Maija Sulonen with working with the exome comparison data. Finally, for all co-workers for creating a pleasant working atmosphere.

Vantaa, February 14, 2011

Henrikki Almusa

Bibliography

Adobe, TIFF v6.0 specification, 2010. Tiff file format specification. Pdf http://partners.adobe.com/public/developer/en/tiff/TIFF6.pdf, checked August 30th, 2010.

Agilent, brochure, 2011. Sureselectxt kits...a surely better workflow solution. Available from http://www.chem.agilent.com/en-US/Search/Library/_layouts/Agilent/PublicationSummary.aspx?whid=70267&liid=1486, checked January 20th, 2011.

Agilent, Exome brochure, 2010. Sureselect human all exon kit. Available from http://www.chem.agilent.com/en-US/Products/reagents/nextgensequencing/targetenrichment/Pages/SureSelectHumanAllExonKit.aspx, checked February 10th, 2010.

Agilent, Exome protocol, 2010. Sureselect human all exon kit protocol. From pdf 'Protocol_SureSelect_HumanAllExon_V1_0_1.pdf'.

D. Blankenberg, G. Von Kuster, N. Coraor, G. Ananda, R. Lazarus, M. Mangan, A. Nekrutenko, and J. Taylor. Galaxy: a web-based genome analysis tool for experimentalists. *Current protocols in molecular biology*, Chapter 19:Unit 19.10.1–21, Jan 2010.

P. J. Cock, C. J. Fields, N. Goto, M. L. Heuer, and P. M. Rice. The sanger fastq file format for sequences with quality scores, and the solexa/illumina fastq variants. *Nucleic acids research*, Dec 16 2009.

T. J. Hubbard, B. L. Aken, S. Ayling, B. Ballester, K. Beal, E. Bragin, S. Brent, Y. Chen, P. Clapham, L. Clarke, G. Coates, S. Fairley, S. Fitzgerald, J. Fernandez-Banet, L. Gordon, S. Graf, S. Haider, M. Hammond, R. Holland, K. Howe, A. Jenkinson, N. Johnson, A. Kahari, D. Keefe, S. Keenan, R. Kinsella, F. Kokocinski, E. Kulesha, D. Lawson, I. Longden, K. Megy, P. Meidl, B. Overduin, A. Parker, B. Pritchard, D. Rios, M. Schuster, G. Slater, D. Smedley,

W. Spooner, G. Spudich, S. Trevanion, A. Vilella, J. Vogel, S. White, S. Wilder, A. Zadissa, E. Birney, F. Cunningham, V. Curwen, R. Durbin, X. M. Fernandez-Suarez, J. Herrero, A. Kasprzyk, G. Proctor, J. Smith, S. Searle, and P. Flicek. Ensembl 2009. *Nucleic acids research*, 37(Database issue):D690–7, Jan 2009.

Illumina, GAB, 2010. Genome analyzer brochure. Pdf `http://www.illumina.com/Documents/products/brochures/brochure_genome_analyzer.pdf`, checked February 10th, 2010.

Illumina, PE sample prep, 2010. Paired-end sequencing sample preparation guide. Pdf 'Paired-End_SamplePrep_Guide_1005063_B.pdf'.

Michal Janitz. *Next-Generation Genome Sequencing*. Wiley-Blackwell, 2008. ISBN 978-3-527-32090-5.

A. R. Jex, R. S. Hall, D. T. Littlewood, and R. B. Gasser. An integrated pipeline for next-generation sequencing and annotation of mitochondrial genomes. *Nucleic acids research*, 38(2):522–533, Feb 2010.

W. J. Kent, C. W. Sugnet, T. S. Furey, K. M. Roskin, T. H. Pringle, A. M. Zahler, and D. Haussler. The human genome browser at ucsc. *Genome research*, 12(6): 996–1006, Jun 2002.

M. Krzywinski, J. Schein, I. Birol, J. Connors, R. Gascoyne, D. Horsman, S. J. Jones, and M. A. Marra. Circos: an information aesthetic for comparative genomics. *Genome research*, 19(9):1639–1645, Sep 2009.

H. Li and R. Durbin. Fast and accurate short read alignment with burrows-wheeler transform. *Bioinformatics (Oxford, England)*, 25(14):1754–1760, Jul 15 2009.

H. Li, B. Handsaker, A. Wysoker, T. Fennell, J. Ruan, N. Homer, G. Marth, G. Abecasis, R. Durbin, and 1000 Genome Project Data Processing Subgroup. The sequence alignment/map format and samtools. *Bioinformatics (Oxford, England)*, 25(16):2078–2079, Aug 15 2009.

M. L. Metzker. Sequencing technologies - the next generation. *Nature reviews.Genetics*, 11(1):31–46, Jan 2010.

A. Nakki, S. T. Kouhia, J. Saarela, A. Harilainen, K. Tallroth, T. Videman, M. C. Battie, J. Kaprio, L. Peltonen, and U. M. Kujala. Allelic variants of il1r1 gene associate with severe hand osteoarthritis. *BMC medical genetics*, 11(1):50, Mar 30 2010.

Nimblegen, Exome brochure, 2010. Seqcap ez exome brochure. Pdf `http://www.nimblegen.com/products/lit/SeqCap_EZ_Exome_LR_Flyer_2009_12_22.pdf`, checked February 8th, 2010.

Nimblegen, Exome protocol, 2010. Nimblegen seqcap ez exome library sr user's guide. Pdf 'SeqCap_UsersGuide_EZ_Exome_SR_v1p1.pdf'.

K. D. Pruitt, J. Harrow, R. A. Harte, C. Wallin, M. Diekhans, D. R. Maglott, S. Searle, C. M. Farrell, J. E. Loveland, B. J. Ruef, E. Hart, M. M. Suner, M. J. Landrum, B. Aken, S. Ayling, R. Baertsch, J. Fernandez-Banet, J. L. Cherry, V. Curwen, M. Dicuccio, M. Kellis, J. Lee, M. F. Lin, M. Schuster, A. Shkeda, C. Amid, G. Brown, O. Dukhanina, A. Frankish, J. Hart, B. L. Maidak, J. Mudge, M. R. Murphy, T. Murphy, J. Rajan, B. Rajput, L. D. Riddick, C. Snow, C. Steward, D. Webb, J. A. Weber, L. Wilming, W. Wu, E. Birney, D. Haussler, T. Hubbard, J. Ostell, R. Durbin, and D. Lipman. The consensus coding sequence (ccds) project: Identifying a common protein-coding gene set for the human and mouse genomes. *Genome research*, 19(7):1316–1323, Jul 2009.

J. Qi, F. Zhao, A. Buboltz, and S. C. Schuster. ingap: an integrated next-generation genome analysis pipeline. *Bioinformatics (Oxford, England)*, 26(1):127–129, Jan 1 2010.

F. Sanger, S. Nicklen, and A. R. Coulson. Dna sequencing with chain-terminating inhibitors. *Proceedings of the National Academy of Sciences of the United States of America*, 74(12):5463–5467, Dec 1977.

S. T. Sherry, M. H. Ward, M. Kholodov, J. Baker, L. Phan, E. M. Smigielski, and K. Sirotkin. dbsnp: the ncbi database of genetic variation. *Nucleic acids research*, 29(1):308–311, Jan 1 2001.

J. T. Simpson, K. Wong, S. D. Jackman, J. E. Schein, S. J. Jones, and I. Birol. Abyss: a parallel assembler for short read sequence data. *Genome research*, 19 (6):1117–1123, Jun 2009.

D. Smedley, S. Haider, B. Ballester, R. Holland, D. London, G. Thorisson, and A. Kasprzyk. Biomart–biological queries made easy. *BMC genomics*, 10:22, Jan 14 2009.

K. V. Voelkerding, S. A. Dames, and J. D. Durtschi. Next-generation sequencing: from basic research to diagnostics. *Clinical chemistry*, 55(4):641–658, Apr 2009.

K. Ye, M. H. Schulz, Q. Long, R. Apweiler, and Z. Ning. Pindel: a pattern growth approach to detect break points of large deletions and medium sized insertions

from paired-end short reads. *Bioinformatics (Oxford, England)*, 25(21):2865–2871, Nov 1 2009.

D. R. Zerbino and E. Birney. Velvet: algorithms for de novo short read assembly using de bruijn graphs. *Genome research*, 18(5):821–829, May 2008.

J. Zimmermann, H. Voss, C. Schwager, J. Stegemann, and W. Ansorge. Automated sanger dideoxy sequencing reaction protocol. *FEBS letters*, 233(2):432–436, Jun 20 1988.

URLs

Alpheus, 2010. http://alpheus.ncgr.org/introduction.jsp, checked February 10th, 2010.

BedGraph Track Format, 2010. http://genome.ucsc.edu/goldenPath/help/bedgraph.html, checked May 25th, 2010.

Broad Institute, IGV. Integrative genomics viewer. Available from http://www.broadinstitute.org/igv, checked August 17th, 2010.

Genome Analysis Toolkit, 2010. http://www.broadinstitute.org/gsa/wiki/index.php/The_Genome_Analysis_Toolkit, checked February 9th, 2010.

GFF specification, 2010. http://www.sanger.ac.uk/resources/software/gff/spec.html, checked February 22th, 2010.

Illumina SNP chip. Human660w-quad v1 dna analysis beadchip kits. http://www.illumina.com/products/human660w_quad_v1_dna_analysis_beadchip_kits.ilmn, checked December 17th, 2010.

Illumina software. Illumina software - genomestudio data analysis software. http://www.illumina.com/software/genomestudio_software.ilmn, checked December 17th, 2010.

NextGENe, 2010. http://www.softgenetics.com/NextGENe.html, checked February 10th, 2010.

Appendix A

VCPipeline configuration

Example of the configuration file for the VCPipeline.

```
[files]
# required
reference sequence = file name of reference genome
sequences 1 = first sequences of paired read
sequences 2 = second sequences of paired read
# optional
# target area = gff file with target areas
# snp details = snp file describing snp information
# exon details = exon position file with gene, transcript and protein ids
# optional output directory
# output directory = directory where the output files go

[tool options]
# required
contig size = minimum size for contig in de-novo assembly
sample name = name of the sample, will appear in gff tracks
minimum depth = minimum depth for variant for snp in variant table
# optional
# if quality ratio smaller assume homozygous call
# quality limit hom = 0.2 (default)
# if quality ratio higher assume reference call
# quality limit het = 0.8 (default)
# minimum number of overlaps to include in circos plot
# circos overlap = 4 (default)
# depth for counting coverage stats
```

```
# coverage depth = minimum depth (default)
# minimum read length = minimum length for read after trimming B from
# read end

[workflow]
#  if there is more than this many reads, then velvet will not be run
de novo max reads = 6000000
# number of threads done with programs using threading (bwa)
threading = 6
```

Appendix B

PCR primers used in post hybridization

These primers were used to verify the results of the PCR. Same set of primers were used with both exome kits.

Table B.1: Verification primers
NSC primers are part of the SeqCapEZ protocols (Nimblegen, Exome protocol, 2010) verification primers while the rest were made for the comparison. The Chr16_cons is a negative target area. That means that it should not show up in the PCR.

Target area	**Direction**	**Sequence**
NSC-0237	forward	CGC ATT CCT CAT CCC AGT ATG
NSC-0237	reverse	AAA GGA CTT GGT GCA GAG TTC AG
NSC-0272	forward	CAG CCC CAG CTC AGG TAC AG
NSC-0272	reverse	ATG ATG CGA GTG CTG ATG ATG
USF1_exon	forward	GTC GAA GCA CGT CAT TGT CC
USF1_exon	reverse	GGA GCT TCG GCA GAG TAA CC
JMY_exon	forward	CCG ATA CAG ACC CTC TAA CAC G
JMY_exon	reverse	CCT CTG ACT CTG GGG ATG CT
Chr16_cons	forward	CTC TTC AGA AAG TGC TAC ACA AGC A
Chr16_cons	reverse	GCT GAG GCT ACC TAG AGA CAT TGA TTA